

Fachlicher Abschlussbericht
Forschungsvorhaben (FKZ: 0327713 B)

AG Turbo COOREFF-T

Verbundprojekt:
CO₂-Reduktion durch Effizienz

Vorhaben-Nr. 1.1.1

„Automatische Optimierung von Verdichterstufen“

Förderkennzeichen: 0327713 B
Vorhabengruppe 1.1: Hocheffiziente Kompressoren für das CO₂-freie Kraftwerk
Zeitraum: 01.07.2005 – 01.07.2008
Projektleitung: Dr.-Ing. E. Nicke
Bearbeiter: Dr.-Ing. C. Voß
Forschungsstelle: DLR Köln, Institut für Antriebstechnik
Industriepartner: MAN TURBO AG, Oberhausen
MTU Aero Engines GmbH, München
Siemens AG, Power Generation, Mülheim

Autoren: C. Voß, E. Nicke
Datum: September 2008

Das diesem Bericht zugrunde liegende Vorhaben wurde mit Mitteln des Bundesministeriums für Wirtschaft und Technologie unter dem Förderkennzeichen 0327713 B gefördert. Die Verantwortung für den Inhalt dieser Veröffentlichung liegt beim Autor.

Kurzfassung

Verdichter in Flugzeugtriebwerken bzw. stationäre Gasturbinenverdichter haben die Aufgabe, das strömende Medium gemäß den Anforderungen aus den thermodynamischen Kreisprozessbetrachtungen möglichst effizient zu verdichten. Des Weiteren sollen die komplexen Strömungsvorgänge in Verdichtergittern, welche aus Verdichtungsstößen resultieren oder durch Wirbel in den Randzonen induziert werden, durch eine gezielte Formgebung der strömungsführenden Bauteile positiv beeinflusst werden. In modernen Entwurfssystemen stehen dafür zahlreiche Designparameter zur Gestaltung des jeweiligen Bauteils zur Verfügung. Diese beeinflussen sowohl den Ringraum als auch die Geometrie der einzelnen Schaufelblätter.

Mathematische Optimierungsstrategien können diesen Designraum gezielt nach optimalen Parametersätzen absuchen um die Gestaltungsfreiheiten in möglichst effektiver Weise zu nutzen. Dabei kommt der Berücksichtigung mehrerer Betriebspunkte im Kennfeld des Verdichters eine besondere Bedeutung zu. Die Tatsache, dass für eine anwendungsorientierte Verdichteroptimierung das Kennfeld und nicht ein einzelner Betriebspunkt Berücksichtigung finden muss, induziert bereits die Notwendigkeit mehrere Zielfunktionen simultan zu betrachten. Eine solche Mehrzieloptimierung wird im Allgemeinen mit dem englischen Begriff Multi-Objective gekennzeichnet.

Das gesamte in diesem Projekt entwickelte Programmpaket besitzt einen modularen Aufbau um die Implementierung von Modulen der Industriepartner sowie die Implementierung zusätzlicher Schnittstellen mit neuen Programmen zu erleichtern. Diese werden sich zukünftig aus der wachsenden Anzahl von Forderungen aus weiteren Fachdisziplinen (z.B. Bauteilmechanik, Fertigung) ergeben. Die integrierte und automatisch überwachte Prozesskette, welche für die aerodynamische Optimierung benötigt wird, besteht aus den Modulen: Schaufelerzeugung, geometrische Restriktionen, Netzerzeugung, CFD, Post-processing.

Hierbei sind unterschiedliche Stömungslöser (Mises, TRACE und Magelan) integriert worden. Die Ersatzmodelle (Kriging, Neuronale Netze) der Simulationsverfahren sind ebenfalls vom eigentlichen Optimierer entkoppelte Module, welche basierend auf einer aktuellen Datenbasis neue viel versprechende Kandidaten erzeugen können, welche dann mittels der eigentlichen Prozesskette bewertet werden.

Als Optimierungsschale wird hier ein asynchroner Multi-Objective Algorithmus verwendet welcher auf der Evolutionsstrategie basiert. Die Evolutionsstrategie bedient sich dem Vorbild der Natur: Besonders erfolgreiche Individuen können Ihre Gene vererben, während erfolglose Individuen dies nicht können. Dadurch entwickelt sich eine Spezies über mehrere Generationen weiter und passt sich der jeweiligen Umgebung immer besser an. Die Vererbung wird innerhalb dieser Strategie mittels mathematischer Operatoren: Mutation, Crossover sowie Differential Evolution nachempfunden und zum Erzeugen neuer Kandidaten genutzt. Eine solche Optimierung hat den Vorteil lokale Minima überwinden zu können und äußerst robust gegenüber den mathematischen Eigenschaften der Zielfunktionen (Stetigkeit, Differenzierbarkeit) zu sein. Der größte Nachteil solcher stochastischen Verfahren liegt in der relativ hohen Anzahl benötigter Simulationen für den notwendigen Optimierungsfortschritt, so dass verschiedene Beschleunigungsmethoden durch Nutzung von Ersatzmodellen wie Kriging und Neuronale Netze in das Optimierungspaket implementiert wurden.

Die komplett dreidimensionale Berechnung der Strömung in einem vielstufigen Axialverdichter ist mit heutigen Rechnerleistungen in einem Optimierungsprozess nicht umsetzbar, da jede einzelne CFD-Berechnung kompletter mehrstufiger Komponenten numerisch zu kostspielig ist. Eine alternative Strategie, zur Berücksichtigung der ganzheitlichen Komponente innerhalb der Optimierung, besteht in der gekoppelten Strömungssimulation.

Dieser setzt sich aus einem 3D-Navier-Stokes-Verfahren zur Lösung der Strömung in einem Teilbereich des Verdichters und einer reibungsfreien aber verlustbehafteten Throughflow-Rechnung des Gesamtverdichters zusammen. Zwischen den beiden Strömungssimulationen müssen Strömungsdaten ausgetauscht werden. Aus der Throughflow-Rechnung werden die Randbedingungen für die 3D-Rechnung extrahiert. Im Gegenzug werden aus den Strömungsdaten der 3D-Rechnung die umfangsgemittelten radialen Verteilungen der Verluste und Abströmwinkel für die Throughflow-Rechnung bestimmt. Hieraus ergibt sich ein iterativer Prozess welcher in konsistenten 2D und 3D Simulationsergebnissen resultiert. Durch die Kopplung der beiden Strömungsrechnungen eröffnen sich weitere Optionen zum Aufbau der Zielfunktionen. Neben den Strömungsgrößen der optimierten Gitter können nun auch die das Gesamtverhalten beschreibenden Kenngrößen in die Zielfunktionsformulierung integriert werden.

Inhaltsverzeichnis

Nomenklatur.....	5
1 Einleitung.....	7
1.1 Problemstellung	7
1.2 Stand der Wissenschaft und Technik	9
1.3 Zielsetzung	12
2 Die automatisierte Prozesskette.....	13
2.1 Geometrieparametrisierung-Bladegenerator	14
2.1.1 Profilierung	
2.1.2 Fädelung	
2.2 Geometrische Restriktionen	22
2.3 Netzerzeugung	23
2.4 Strömungslösung	27
2.4.1 Q3D-Euler Grenzschicht MISES	
2.4.2 3D Navier Stokes TRACE	
2.4.3 Gekoppelter Löser TRACE-MAGELAN	
2.5 Strukturanalyse	33
2.6 Parameter/Zielfunktionen	
3. Das Optimierungspaket <i>AutoOpti</i>.....	37
3.1 Aufbau und Parallelisierung.....	38
3.1.1 Vererbungsoperatoren	
3.1.2 Datenbasis	
3.1.3 Nebeneingang	
3.1.4 Paretorang	
3.1.5 DOE	
3.2 Metamodelle.....	42
3.2.1 Kriging	
3.2.2 Neuronale Netze	
3.2.1 Ersatzmodell-Mittelung und Optimierung	
4 Zielfunktionsformulierung.....	53
5. Optimierung mittels Trace/Calculix: gegenläufiger Fan	54
6 Optimierung mittels des gekoppelten Strömungslösers	60
7. Verwertung	62
8 Zusammenfassung	63
Literaturverzeichnis	63

Nomenklatur

Abkürzungen

CFD	Computational Fluid Dynamics
FEM bzw FE	Finite Element Method
NS	Navier-Stokes
MO	Multi-Objective
SO	Single-Objective
ES	Evolutionsstrategie
DOE	Design of Experiments
LE bzw VK	Vorderkante
TE bzw HK	Hinterkante
EO	Engine-Order
1F bzw 2F	Biegung
1T bzw 2T	Torsion

Schaufel- Profilparametrisierung

β_{ST}	Staffelungswinkel
β_{LE}	Vorderkanten Metallwinkel
$d\beta_{LE}$	Vorderkanten Keilwinkel
R_{LE}	Vorderkanten-Radius
$A2B_{LE}$	Vorderkanten-Halbachsenverhältnis
$Asym_{LE}$	Asymmetriefaktor Vorderkante
LE_{FF}	Vorderkanten-Formfaktor
c_{ax}	axiale Länge
HP	Hilfspunkt zur Konstruktion der VK/HK
alternative Druckseite:	
S_Dmax	Lage der maximalen Profildicke
$Dmax$	maximale Profildicke
Fill	Füllungsgrad
FillFront	Füllungsgrad vorne
TP_front	Übergangslage $f_1 \rightarrow f_2$
$dFrontArea$	prozentuale Flächenvergrößerung alternative Druckseite
Fädelung	
x-shift, θ -shift, m-shift, S-shift	Koordinaten der Fädellinie

B-Spline:

$C(u)$	B-Spline Kurve
$N_{i,p}(u)$	B-Spline Basisfunktion
\bar{P}	B-Spline Kontrollpunkt
U bzw. \tilde{U}	B-Spline Knotenvektor
n	Anzahl Kontrollpunkte -1
m	Anzahl Knoten-1
p	Grad des B-Splines

Metamodelle:

s^2 bzw. σ^2 bzw. Var	Varianz
$\hat{\beta}$	Erwartungswert Kriging
Θ, p, λ	Hyperparameter Kriging
R_{ij} bzw. \tilde{R}	Korrelationsmatrix
Cor	Korrelationsfunktion
α, β	Hyperparameter Neuronale Netze
ϕ bzw. j	Standardnormalverteilung
F, F_D, F_W	Fehlerfunktion
A, H	Hessematrix von F, F_D
w, w_i, \tilde{w}	Gewichte
W	Anzahl Gewichte
M	Anzahl Trainingsmuster
μ bzw. $E[]$	Erwartungswert
X	Zufallsvariable

Physikalische Größen und Kennzahlen

T, T_t	Temperatur/Totaltemperatur
P, P_t	Druck / Totaldruck
π, π_t	Druckverhältnis / Totaldruckverhältnis
$Ma; M_{is}$	Machzahl
η	Wirkungsgrad
ω	Verlustbeiwert
\dot{m}	Massenstrom
ρ	Dichte
σ	Spannung
α, β	Strömungswinkel

1 Einleitung

Die Verdichterauslegung ist eine multidisziplinäre Aufgabe in der unterschiedlichste Anforderungen bewertet und gegeneinander abgewogen werden müssen. Diese ergeben sich zum einen aus den unterschiedlichen Disziplinen wie Aerodynamik, Strukturmechanik, Aeroelastik usw. Zum anderen stellt jede einzelne dieser Disziplinen unterschiedliche Anforderungen an das Verdichterdesign. So muss aus aerodynamischer Sicht in allen relevanten Betriebspunkten Rücksicht genommen werden auf: Massenstrom, Wirkungsgrad, Totaldruckverhältnis, Pumpgrenze.

Die Verdichterauslegung stellt daher immer einen Kompromiss aus den unterschiedlichen, größten Teils konträren Anforderungen dar.

Eine signifikante Steigerung der Leistung und Effizienz moderner Axialverdichter erscheint nur möglich, wenn sowohl möglichst viele geometrische Parameter als auch physikalische Ergebnisse in der Auslegungsphase berücksichtigt werden und dabei zusätzlich zahlreiche Nebenbedingungen eingehalten werden.

In einer Mehrzieloptimierung können solche Aspekte über die Formulierung der unterschiedlichen Zielfunktionen und Restriktionen grundsätzlich erfasst werden. Im hier beschriebenen Projekt wird mit der Evolutionsstrategie eine Mehrzieloptimierung gewählt, welche keine weiteren Anforderungen an die Zielfunktionen stellt und somit auch die Behandlung z.B. nicht konvergenter Simulationen und somit unstetiger Zielfunktionen erlaubt.

1.1 Problemstellung

Die vielfältigen Anforderungen an moderne Verdichter ergeben sich u.a. aus den unterschiedlichen Betriebspunkten eines Verdichters, die sich nicht nur in den aerodynamischen Randbedingungen sondern auch durch die jeweils erwünschte Verdichterperformance im betrachteten Betriebspunkt unterscheiden. Gleichzeitig sind Forderungen zur mechanischen Festigkeit, der Lebensdauer sowie zu Aspekten der Herstellbarkeit, der Reparierbarkeit usw. zu beachten.

Ziel des hier beschriebenen Vorhabens war die Entwicklung einer Technologie zur automatisierten aerodynamischen Optimierung von dreidimensionalen Verdichterschaufeln, bzw. des gesamten die Strömung führenden Kanals, unter Berücksichtigung mehrerer Betriebspunkte und physikalischer Bewertungskriterien aus dem gesamten Arbeitsbereich des Verdichters.

Die Bearbeitung dieses Vorhabens fand in enger Kooperation mit der Universität Kassel und dem dort bearbeiteten Vorhaben 1.1.2 (Förderkennzeichen 0327720A) statt. In einer detaillierten Abstimmung wurden zwischen den beiden Forschungsstellen die bereits vorhandenen Programmkomponenten zur automatischen Optimierung ausgetauscht und die vom DLR bis zu diesem Zeitpunkt durchgeführten ersten Untersuchungen zum Netzeinfluss auf das CFD-Ergebnis wurden der Universität Kassel zur Verfügung gestellt.

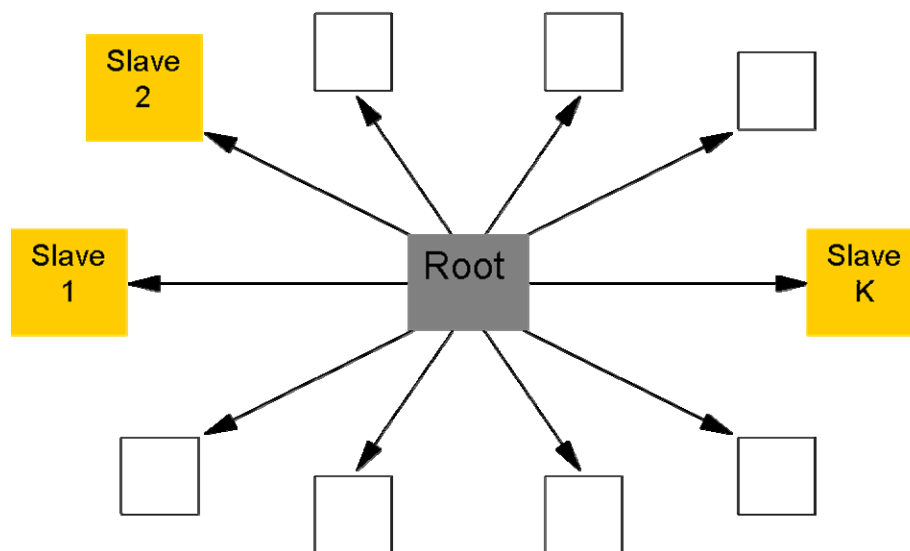
Außerdem erfolgten Festlegungen bezüglich der genutzten Programmiersprachen, Scriptsprachen und Rechnerstrukturen sowie der 3D-Schaufelbeschreibung, der zu nutzenden Strömungslöser und Netzgeneratoren.

Ziel dieses initialen Austausch- und Abstimmungsprozesses war die Verwendung und Weiterentwicklung einer gemeinsamen Optimierungsplattform für alle aerodynamischen Anwendungen von Throughflow-Optimierungen der Vorauslegung bis hin zur vollen 3D Navier-Stokes Optimierung von Verdichterstufen.

Die Automatisierung der zur aerodynamischen Optimierung erforderlichen Prozesskette (siehe Kapitel 2) stellte einen ersten Arbeitsschwerpunkt innerhalb dieses Projektes dar. Diese Prozesskette beinhaltet sowohl die Geometrieparametrisierung als auch die Netzerzeugung und die automatische Durchführung und Überwachung der CFD-Rechnungen (Q3D, 3D oder gekoppelt 2D-3D). Jedes dieser Module muss fehlerfrei und ohne manuellen Eingriff arbeiten und ggf. (nicht konvergierende Simulation, unsinnige Wahl von Designparametern, usw.) kontrolliert und automatisch abgebrochen werden.

Die nach dem Vorbild der Natur entwickelte Evolutionsstrategie (siehe Kapitel 3.1) ist eine stochastische Suchmethode und benötigt weder Gradienteninformationen noch Stetigkeitsanforderungen. Die Eindeutigkeit der Zielfunktion, d.h. die Forderung, dass die Zielfunktion die Forderung einer mathematischen Abbildung erfüllt und somit jedem Datensatz genau einen Zielfunktionswert bzw. -vektor zuordnet, ist hinreichend für die Anwendbarkeit dieser Optimierungsstrategie.

Parallelisierbare Komponenten der Evolutionsstrategie sind neben der Evaluierung der Zielfunktion die Generierung von Zufallszahlen, die Rekombination der Eltern und die Mutation der Parameter. Der Rechenzeitbedarf der letztgenannten Komponenten ist im vorliegenden Fall jedoch gegenüber einer Zielfunktionsauswertung zu vernachlässigen, welche für die aerodynamische Optimierung die zeitintensive CFD-Simulationen benötigt. Dies legt eine sternförmige Prozessstruktur (vergleiche Bild 1.1) nahe, in welchen der Root-Prozess für die eigentliche Optimierung und die Slave-Prozesse für die Auswertung der Zielfunktionen verantwortlich sind. Als Software-Plattform für die Parallelisierung bieten sich u. a. die Programmpakete Parallel Virtual Machine (PVM) und Message Passing Interface (MPI) an, die verschiedene Rechner eines Netzwerks zu einem einzigen, virtuellen Parallelrechner mit verteiltem Speicher koppeln. Aufgrund der Vielzahl benötigter Zielfunktionsevaluierungen und des großen numerischen Aufwandes für jede einzelne CFD-Rechnung ist eine solche Parallelisierung für eine praxistaugliche Optimierungsumgebung unerlässlich. Im Rahmen dieses Vorhabens wurde MPI als Programmbibliothek zur Parallelisierung gewählt.



Vergleichende Untersuchungen und Strategien zur Reduktion der Zielfunktionsaufrufe bildeten einen weiteren Schwerpunkt der Arbeiten (Kapitel 3.2). Das größte Potential zur Reduzierung der aufwendigen physikalischen Simulationen bieten mathematische Methoden zur Approximation der Zielfunktionen wie sie durch Kriging und neuronale Netze gegeben sind. Diese haben zur Aufgabe eine, während der Optimierung entstandene, Datenbasis möglichst *sinnvoll* zu approximieren. Dies ist insbesondere in hochdimensionalen Räumen eine anspruchsvolle Aufgabe. Zusätzlich wurden an der Universität Kassel (Förderkennzeichen 0327720A) innerhalb des

Vorhabens 1.1.2 polynomiale Antwortflächen untersucht, welche in höheren Dimensionen jedoch nur schwer einsetzbar erscheinen.

Da die Optimierung einzelner Komponenten vielstufiger Turbomaschinen Wechselwirkungen mit den übrigen (nicht optimierten) strömungsführenden Bauteilen bedingen und somit das Gesamtverhalten der Komponente beeinflussen, ist es sinnvoll, diese gegenseitige Beeinflussung möglichst weitgehend zu berücksichtigen. Eine Kopplung der 3D Simulation (mit dem DLR inhouse code TRACE) mit einem 2D-Meridianströmungsverfahren wurde entwickelt um den Einfluss eines veränderten Gitters auf die benachbarten Schaufelreihen und den Gesamtverdichter bereits während der Optimierung zu berücksichtigen. Diese Vorgehensweise ist dem Umstand geschuldet, dass reine 3D Optimierungen von ganzen Komponenten (vieltufige Verdichter) den Leistungsbereich modernster Hochleistungsrechencluster heutzutage noch sprengen.

Neben der Entwicklung und dem Aufbau des modular aufgebauten Optimierers mitsamt dazu gehöriger Prozesskette, bildeten die Untersuchungen zum Aufbau der Zielfunktionen (Kapitel 4) unter Einschluss konstruktiver und mechanischer Gesichtspunkte und Restriktionen einen weiteren Arbeitsschwerpunkt innerhalb dieses Projektes.

Das Problem der Verdichteroptimierung besteht allgemein formuliert darin, den Vektor der Designparameter so zu bestimmen, dass die Zielfunktionen minimiert werden (Maximierungsprobleme werden durch Negation in Minimierungsprobleme transformiert). Zusätzlich sind Nebenbedingungen zu erfüllen, die die Menge der zulässigen Lösungen einschränken. Als vordringliches Ziel der aerodynamischen Optimierung von Verdichterschaukeln kann z.B. die Minimierung der Verluste im aerodynamischen Designpunkt betrachtet werden. Ein weiteres wichtiges Optimierungsziel ist die Verbesserung oder Zumindest die Beibehaltung des Arbeitsbereichs auf unterschiedlichen Drehzahllinien. Aerodynamische Restriktionen können sich aus geforderten Massenströmen oder Druckverhältnissen ergeben.

1.2 Stand der Wissenschaft und Technik

Eine Optimierung soll innerhalb dieser Arbeit als Suche nach denjenigen Punkten in einem Designraum (oder Definitionsbereich) verstanden werden, welche eine Zielfunktion, oder eine beliebige Linearkombination unterschiedlicher Zielfunktionen, minimieren. Es gibt viele Möglichkeiten der Minimumsuche auf mathematischen Funktionen. Unterteilen lassen sich diese Optimierungsstrategien beispielsweise in deterministische und stochastische Verfahren.

Deterministische Verfahren

Sie heißen deterministisch, weil sie immer ein identisches Verhalten zeigen, wenn man sie am gleichen Startpunkt beginnen lässt. Die Art die Suchrichtung zu bestimmen teilt diese Verfahren in zwei große Gruppen ein. Man spricht von indirekten Verfahren, wenn dazu Informationen der partiellen Ableitung der Zielfunktion benötigt werden und von direkten, wenn stattdessen Zielfunktionswerte unter Anwendung von Heuristiken genügen. Der prinzipielle Aufbau ist bei allen deterministischen Verfahren ähnlich:

Ausgehend vom Vektor des Startpunkts bzw. des zuletzt ermittelten Punkts werden Suchrichtung und Schrittlänge ermittelt, um daraus einen neuen Punkt zu berechnen. Dieser wird dann bewertet und ggf. ein Abbruchkriterium überprüft. Die Verfahren unterscheiden sich vor allem in der Ermittlung der Suchrichtung und der Schrittlänge.

Die indirekten Verfahren setzen erstens voraus, dass die Zielfunktionen an einem Punkt analytisch auswertbar sind und dass sie zweitens zumindest in einer Umgebung des aktuellen Punktes auch

differenzierbar ist. Bei vielen Aufgabenstellungen, die z.B. auf Simulationsmodelle zur Bestimmung des Wertes der Qualitätsfunktion zurückgreifen wie in diesem Projekt, ist dies nicht der Fall. Wenn diese Voraussetzungen dagegen erfüllt sind, spricht die vergleichsweise geringe Anzahl an benötigten Funktionswertberechnungen für den Einsatz der indirekten Verfahren. Zu ihren bekanntesten Vertretern gehören:

- das Gradientenverfahren,
- das konjugierte Gradientenverfahren und
- das Quasi-Newton-Verfahren samt Varianten.

Da die direkten Verfahren ohne eine Ableitung der Zielfunktionen auskommen und lediglich eine stückweise Stetigkeit verlangen, sind sie breiter anwendbar. Zu ihnen zählen die Pattern- und die Polyeder-Strategien. Die Pattern-Strategien nutzen Erkundungsschritte und den Differenzenquotienten zur Bestimmung der Richtung des nächsten Suchschrittes. Dazu gehören

- das Verfahren nach Hooke und Jeeves,
- der Powell Algorithmus

Die Polyeder-Strategien nutzen im Gegensatz zu den bisher besprochenen Algorithmen die Informationen von mehreren Punkten im Suchraum gleichzeitig. Diese Punkte bilden einen Polyeder im Raum. Zu ihnen gehören

- der Simplex-Algorithmus nach Nelder und Mead
- der Complex-Algorithmus als Variante des Simplex-Verfahrens für beschränkte Aufgaben.

Stochastische Verfahren

Im Unterschied zu deterministischen Verfahren bestimmt hier zum Teil der Zufall das Ergebnis einer Optimierung. Beispiele für stochastische Optimierungsverfahren sind:

- Simulated annealing
- Genetische Algorithmen
- Evolutionäre Algorithmen
- Ameisenkolonie Algorithmus

Die stochastischen Verfahren benötigen im Gegensatz zu deterministischen Methoden nur die Zielfunktionswerte. Die Abstiegsrichtung wird über Tastschritte in der näheren Umgebung ermittelt. Nebenbedingungen werden über Lagrange'sche Multiplikatoren oder über Strafterme in der Zielfunktion bzw. Aufschläge auf den Paretorang berücksichtigt. In vergleichenden Studien zum Einsatz von Optimierungsverfahren für die aerothermodynamische Auslegung (siehe z.B. [Müller-Töws 2000], [Shahpar 2000] und [Sasaki et al. 2001]) haben sich stochastische Verfahren gegenüber den deterministischen Verfahren bei großen Dimensionen des Suchraumes als besser geeignet erwiesen. Bei den deterministischen Verfahren besteht die Gefahr, dass die Optimierung in einem lokalen Minimum stagniert, während stochastische Methoden aufgrund der Zufallssteuerung aus diesen lokalen Minima wieder herausfinden können. Zudem bieten diese Verfahren einen hohen Parallelisierungsgrad.

Im Folgenden soll nun etwas näher auf die Evolutionsstrategie eingegangen werden welche getreu dem Motto: "Die Natur ist der beste Baumeister!" vorgeht. Mehr noch, die Natur liefert eine Fülle von Beispielen für die unterschiedlichsten Arten, die hervorragend an die jeweiligen Lebensbedingungen angepasst sind. Da die Herausbildung dieser Eigenschaften ohne theoretisches Wissen erfolgte, muss in der Natur ein Wirkmechanismus enthalten sein, der z.B. den Vögeln das Fliegen gestattet ohne dass sie etwas über Aerodynamik wissen. Seit Darwin ist dieser Mechanismus auch bekannt: Variation bei der Informationsweitergabe durch Vererbung und Selektion der Besten (survival of the fittest).

Evolutionäre Algorithmen (EA) versuchen nun, die Prinzipien der natürlichen Evolution im Computer nachzubilden, um damit Problemlösungen quasi zu züchten anstatt sie analytisch zu berechnen. Entsprechend dem natürlichen Vorbild stellen sie kaum Voraussetzungen an den Suchraum. Die Zielfunktion muss weder stetig noch differenzierbar sein, es wird lediglich gefordert, dass sie so beschaffen ist, dass sie eine Suche in irgendeiner Weise zum Ziel führen kann.

Bild 3.1 zeigt den prinzipiellen Ablauf Evolutionärer Algorithmen. Ausgehend von einer Startpopulation, die zufällig oder mittels eines **Design of Experiments** erzeugt sein kann, erfolgt für alle bewerteten Individuen der Zyklus: Gütebestimmung (über Paretorang), Speicherung des Individuums in einer Datenbasis, Elternwahl, Erzeugung von Nachkommen - Bewertung.

Die EAs entstanden parallel: Die Evolutionsstrategie von Rechenberg [Rechenberg 1994] und weitergeführt von Schwefel [Schwefel 1995] in Deutschland einerseits und die Genetischen Algorithmen von Holland [Holland 1975] sowie die der Evolutionsstrategie in mancher Hinsicht ähnliche Evolutionäre Programmierung von Fogel [Fogel 1994] in den USA. Neben diesen Grundformen Evolutionärer Algorithmen gibt es noch viele Varianten wie z.B. die Genetische Programmierung von Koza [Koza 1992].

Nissen [Nissen 1997] gibt einen guten Überblick über die wichtigsten Ausprägungen Evolutionärer Algorithmen.

Ein Beispiel einer angewandten aerodynamischen Optimierung mit einem genetischen Algorithmus wurde von Trigg et al. 1997 [Trigg et al. 1997] veröffentlicht. Ziel war die Verlustreduzierung von Industriedampfturbinen. Unter Einsatz des Dawes-Codes konnten zweidimensionale Profilschnitte aerodynamisch optimiert werden. Die Profile sind dabei durch siebzehn Geometrie Größen parametrisiert worden.

Der optimierte Entwurf transsonischer Profile mit Hilfe genetischer Algorithmen ist in [Oyama 2000] dargestellt. Die aerodynamische Bewertung erfolgt hier mit einem 3D-Navier-Stokes-Code. Die extrem zeitaufwändigen Strömungsberechnungen wurden auf einem hochleistungsfähigen Parallelrechner, dem "Numerical Wind Tunnel" (NAL, Japan) durchgeführt.

[Dennis 1999] kombinierten einen genetischen Algorithmus mit einem SQP-Verfahren (Gradienten basiert). Durch die effizientere Behandlung der Nebenbedingungen konnten Zielfunktionsaufrufe im nicht sinnvollen Lösungsbereich reduziert werden. Sie stellen Ergebnisse für zweidimensionale aerodynamische Optimierungen mit einem Navier-Stokes-Löser auf unstrukturierten Netzen vor.

Pierret [Pierret und van den Braembussche 1999] hat sich in seiner Arbeit sehr umfassend mit dem Einsatz neuronaler Netze befasst. Er stellt einige erfolgreiche zweidimensionale aber auch schon erste dreidimensionale aerodynamische Optimierungen mit dieser Technik vor.

Eine ebenfalls sehr umfassende Arbeit der Anwendung der Kombination von genetischen Algorithmen und neuronalen Netzen zur zweidimensionalen aerodynamischen Optimierung von Profilen wurde von [Uelschen 2000] vorgestellt

Zahlreiche Veröffentlichungen beschäftigen sich mit der Verwendung Neuronaler Netze als Antwortflächen in der Turbomaschinenauslegung, z.B. in der Entwicklung von Triebwerkskomponenten wie bei [Vaidyanathan et al. 2000]. Einen umfassenden Überblick geben [Shyy et al. 2001].

Zur Maximierung des Wirkungsgrades einer mehrstufigen Axialturbine verwenden [Petrovic et al. 2000] eine Hybridmethode, die im Programmverlauf zwischen verschiedenen deterministischen und stochastischen Optimierungsalgorithmen hin- und herwechselt. Optimiert werden die Gehäusekontur sowie die Ein- und Austrittsbedingungen jedes Gitters.

Basierend auf diesen Vorarbeiten wurde im vorliegenden Projekt ein Optimierer entwickelt, welcher die aerodynamische 3D Optimierung von Verdichterstufen erlaubt und die Einbindung weiterer physikalischer Disziplinen ermöglicht. Die schnell wachsenden Rechnerkapazitäten

erlauben heutzutage die Nutzung und Einführung dieser, zu Beginn des Projektes noch sehr visionären anmutenden 3D Optimierungsläufe, in die industrielle Praxis.

1.3 Zielsetzung

Ziel des Vorhabens war es ein Programmpaket zu erarbeiten welches die automatisierten aerodynamische Optimierung von dreidimensionalen Verdichterschaufeln, bzw. des gesamten die Strömung führenden Kanals, unter Berücksichtigung mehrerer Betriebspunkte aus dem Arbeitsbereich des Verdichters ermöglicht.

Dabei sollte der Schwerpunkt des Vorhabens in der aerodynamischen Optimierung von Axialverdichtern unter Einbeziehung des Verhaltens des Gesamtverdichters liegen.

Mit dem hier beschriebenen Projekt sollte eine weitere Leistungssteigerung bei der Entwicklung von Turbomaschinen mit der Verbesserung des räumlichen transsonischen und supersonischen Schaufelentwurfes in der industriellen Praxis erreicht werden. Dabei wird der Ingenieur von zeitintensiven Studien befreit und Ihm werden Wege aufgezeigt bekommen, das Auslegungsziel schnell und zielgenau zu erreichen.

Mit diesen leistungsgesteigerten bzw. wirkungsgradoptimierten Komponenten werden notwendige Voraussetzungen für den Aufbau von Kraftwerksprozessen mit maximalem Systemwirkungsgrad und minimiertem CO₂-Ausstoß geschaffen.

Des Weiteren sollten Kriterien und Vorschläge für die Aufstellung der Zielfunktion – der Kernpunkt jeder programmtechnisch ausgeführten Optimierung – erarbeitet werden. Der Formulierung der Zielfunktion kommt eine zentrale Bedeutung zu, da durch eine geeignete Auswahl die „Entwicklungsrichtung“ und Qualität des Ergebnisses gesteuert und auch der Rechenaufwand beeinflusst wird.

Mit dem modularen Aufbau des Programms sollte außerdem die Möglichkeit geschaffen werden, eine wachsende Anzahl von Forderungen aus weiteren Fachdisziplinen in die automatische Optimierung einzubeziehen.

Mit der automatischen Optimierung von Turbomaschinen können, in Erweiterung zur klassischen Auslegungsmethodik:

- Ideen für neue bisher nicht in Betracht gezogene Oberflächenformen - oder auch Grenzen z.B. der Belastung gefunden,
- vorhandene Beschaukelungen verbessert und
- durch die Integration in die industrielle Prozesskette Entwicklungszeit und –Kosten gespart werden.

2. Die automatisierte Prozesskette

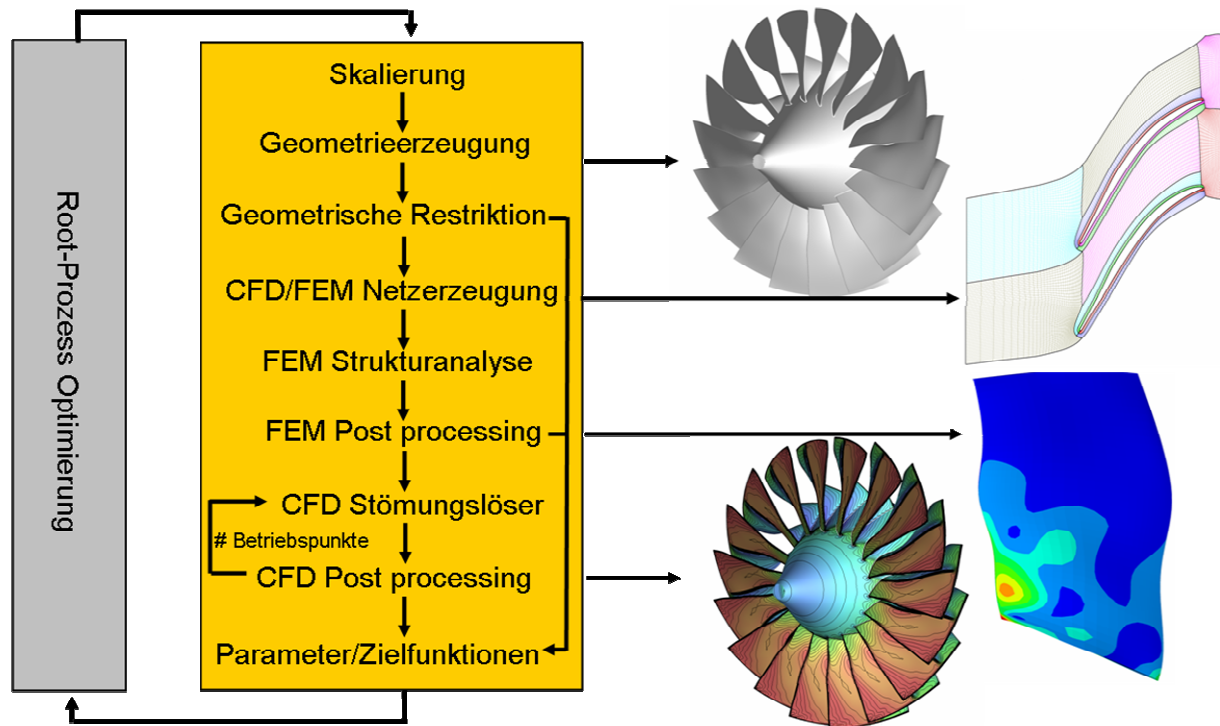


Bild 2.1: Automatisierte Prozesskette der aerodynamischen Verdichterauslegung

Bevor man die aerodynamische Verdichterauslegung mit einer Optimierungsstrategie zu einem automatisierten Auslegungstool verbinden kann, muss die benötigte Prozesskette (siehe Bild 2.1) einen Automatisierungsgrad erreichen, welcher ermöglicht, dass im Prozessablauf auf jeglichen manuellen Eingriff verzichtet werden kann und für alle auftretenden Parametersätze konvergente Lösungen oder definierte Fehlermeldungen erzeugt werden.

Im Folgenden wird nun auf die einzelnen Module dieser Prozesskette eingegangen. Diese Module mitsamt den Schnittstellen untereinander, wurden im Institut für Antriebstechnik des DLR in Köln in Hinblick auf den Einsatz im Optimierungsprozess entwickelt.

Skalierung:

Jeder Parameter der innerhalb der Optimierung variiert werden darf, kann dies nur in definierten Grenzen tun:

$$\alpha \in [\alpha_u, \alpha_o]$$

Innerhalb des Optimierers wird dann die normierte Zahlendarstellung gewählt:

$$\bar{\alpha} = \frac{\alpha - \alpha_u}{\alpha_o - \alpha_u} \in [0,1] \text{ da diese die unterschiedlichen Wertebereiche vergleichbar macht.}$$

Die Prozesskette empfängt diese normierte Variable, welche dann im Modul **Skalierung** (Siehe Bild 2.1) in Ihren Ausgangswert zurück transformiert werden muss:

$$\alpha = \alpha_u + \bar{\alpha}(\alpha_o - \alpha_u)$$

Mittels dieser freien Variablen und zusätzlicher nicht für die Optimierung frei gegebenen statischer Variablen, wird dann eine Struktur belegt, welche die Gesamtgeometrie des zu evaluierenden Verdichters enthält.

Dieser Verdichter durchläuft dann die benötigten Programm-Module, die in Bild 2.1 schematisch dargestellt sind:

2.1 Geometrieerzeugung-Bladegenerator

Das Programm zur Geometrieerzeugung des Verdichters ist mit dem Optimierungsprogramm am engsten verwoben: Die gewählte Parametrisierung der Geometrie bestimmt die Variationsmöglichkeiten innerhalb der Optimierung und ein vollständiger Verdichterdatensatz bzgl. dieser Parametrisierung muss zu Beginn der Prozesskette definiert sein.

Zuerst wird der Naben und Gehäuseverlauf festgelegt. Danach wird für alle Schaufelreihen nacheinander das Programm zur Schaufelerzeugung durchlaufen, welches zu einem großen Teil innerhalb des hier beschriebenen Projektes entwickelt wurde.

Das Verfahren zur Schaufelgeometrieerzeugung, der sog. Bladegenerator, ist dabei in Hinblick auf die Optimierung so ausgelegt worden, dass folgende Kriterien erfüllt werden:

- Kompatibilität mit anderen Programmen (z.B. FE-Modell, CAD, Netzgenerator)
- Grosse Freiheit in der gezielten Gestaltung von Krümmungsverteilungen zur Beeinflussung von Stoß-Grenzschicht Interaktionen.
- Möglichkeit zur globalen und insbesondere zur gezielten lokalen Beeinflussung der Schaufelgeometrie.
- Beibehaltung von typischen Designparametern (wie z.B. Staffelungswinkel, Keilwinkel)

Der Aufbau der Schaufel erfolgt in zwei wesentlichen Schritten:

Profilierung und Fädelung die in den folgenden Abschnitten kurz beschrieben werden.

2.1.1 Profilierung

Die Beschreibung der Profile erfolgt im zweidimensionalen (m', θ) Koordinatensystem (siehe hierzu Bild 2.5 und [Drela 1998]) da dieses eine winkeltreue Transformation zwischen dem 2-dimensionalen Entwurfsraum und dem finalen 3-dimensionalen Koordinatensystem erlaubt.

In diesem 2-dimensionalen Entwurfsraum werden für jedes Konstruktionsprofil 4 Kurven in der Form von B-Splines für Vorderkante, Hinterkante, Saugseite und Druckseite entworfen, welche anschließend zu einer einzigen B-Spline Kurve verbunden werden (Zu B-Splines siehe [Piegl und Tiller 1997]).

Im 2-dimensionalen Entwurfsraum werden die Profile im Wesentlichen durch die folgenden geometrischen Parameter definiert:

1. Staffelungswinkel	β_{ST}
2. Axiale Länge	c_{ax}
3. Vorderkanten Metallwinkel	β_{LE}
4. Vorderkanten Keilwinkel zur Saugseite	$d\beta_{LE_SS}$
5. Vorderkanten Keilwinkel zur Druckseite	$d\beta_{LE_PS}$
6. Vorderkantenradius	R_{LE}
7. Halbachsenverhältnis Vorderkante	$A2B_{LE}$
8. Asymmetriefaktor Vorderkante	$Asym_{LE}$
9. Formfaktor Vorderkante	LE_{FF}
10. Hinterkanten Metallwinkel	β_{TE}
11. Hinterkanten Keilwinkel zur Druckseite	$d\beta_{TE_SS}$
12. Hinterkanten Keilwinkel zur Druckseite	$d\beta_{TE_PS}$
13. Hinterkantenradius	R_{TE}
14. Halbachsenverhältnis Hinterkante	$A2B_{TE}$

Hinzu kommen noch weitere Parameter die den detaillierten Verlauf von Saug- und Druckseite des Profils bestimmen (s.u. B-Spline Kontrollpunkte).

Hierbei sind die beiden ersten Parameter Staffelungswinkel und axiale Länge für die lineare Transformation vom gestaffelten (m', θ) ins ungestaffelte ($\underline{m}', \underline{\theta}$)-System verantwortlich (Siehe Bild 2.2).

Die anderen Parameter 3.-14. in obiger Liste finden im ungestaffelten ($\underline{m}', \underline{\theta}$)-System Verwendung, in welchem der Vorderkantenpunkt des Profils im Ursprung liegt und der Hinterkantenpunkt durch (1,0) definiert ist (Siehe Bild 2.2).

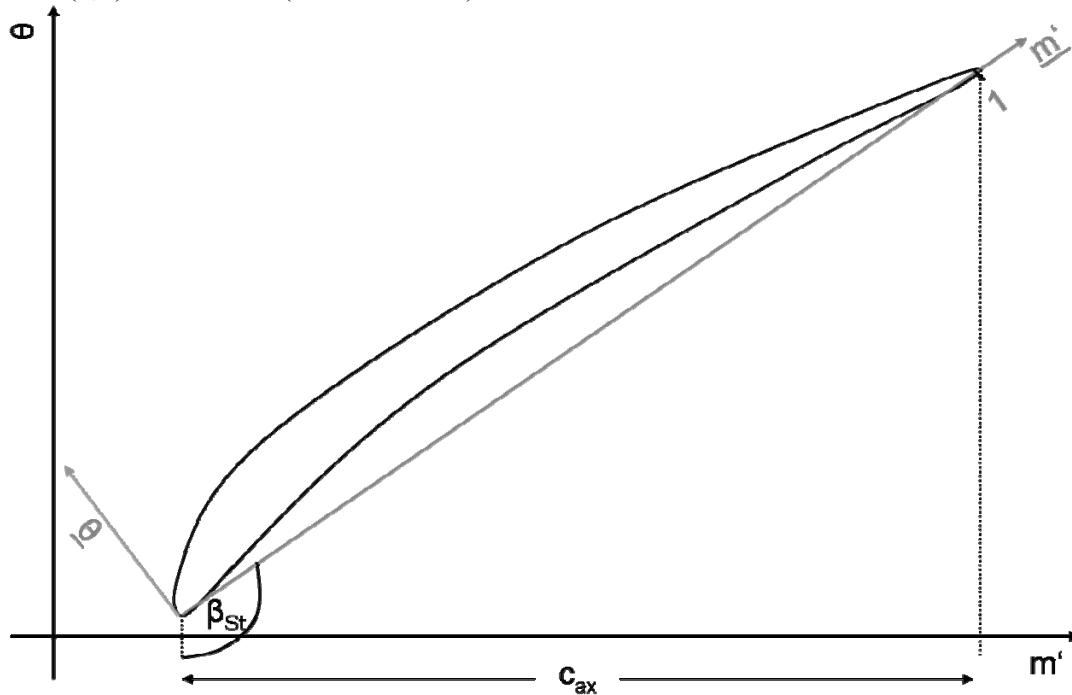


Bild 2.2: Ungestaffelte Koordinaten.

Im nachfolgenden wird nun die Erzeugung der Profile im ungestaffelten System beschrieben an welche sich dann nur noch die Drehstreckung mittels der Parameter Staffelungswinkel und axiale Länge nebst einer evtl. Verschiebung des Koordinatenursprungs anschließt.

Mit den Angaben der Profilparameter 3-14 werden folgende Punkte und Vektoren aus Bild 2.3 bestimmt:

Der Vorderkantenpunkt P_{VK} liegt im Ursprung des Koordinatensystems.

V_{VK} zeigt in Richtung $\beta_{LE} - \beta_{St}$

V_{VK_SS} zeigt in Richtung $\beta_{LE} - \beta_{St} + d\beta_{LE_SS}$

V_{VK_PS} zeigt in Richtung $\beta_{LE} - \beta_{St} - d\beta_{LE_DS}$

P_{VK_SS} und P_{VK_PS} ergeben sich aus dem Vorderkantenradius R_{LE} , dem Halbachsenverhältnis $A2B_{LE}$, der initial elliptisch angenommenen Vorderkante (mit den Halbachsen a und b) sowie dem Vorderkantenwinkel und dem Asymmetrie- und Formfaktor der Vorderkante.

Der Beginn der Konstruktion erfolgt mit der Annahme:

$Asym_{LE} = LE_{FF} = 0$

Dann ergeben sich P_{VK_SS} und P_{VK_PS} sofort aus der Richtung von V_{VK} , dem Vorderkantenradius R_{LE} , und dem Halbachsenverhältnis $A2B_{LE}$. Danach werden die Punkte P_{VK_SS} und P_{VK_PS} auf Ihrer Verbindungsgeraden um den Vektor $Asym_{LE}$ ($P_{VK_SS} - P_{VK_PS}$) verschoben. Wobei die „Nulllage“ der Punkte P_{VK_SS} und P_{VK_PS} so gewählt ist, dass sich die Geraden mit den Richtungsvektoren $P_{VK_SS} + u \cdot V_{VK_SS}$ und $P_{VK_SS} + v \cdot V_{VK_PS}$ im Ursprung schneiden.

Mit der Hinterkante wird entsprechend verfahren, wobei hier jedoch auf die Freiheit der asymmetrischen Gestaltung verzichtet wird.

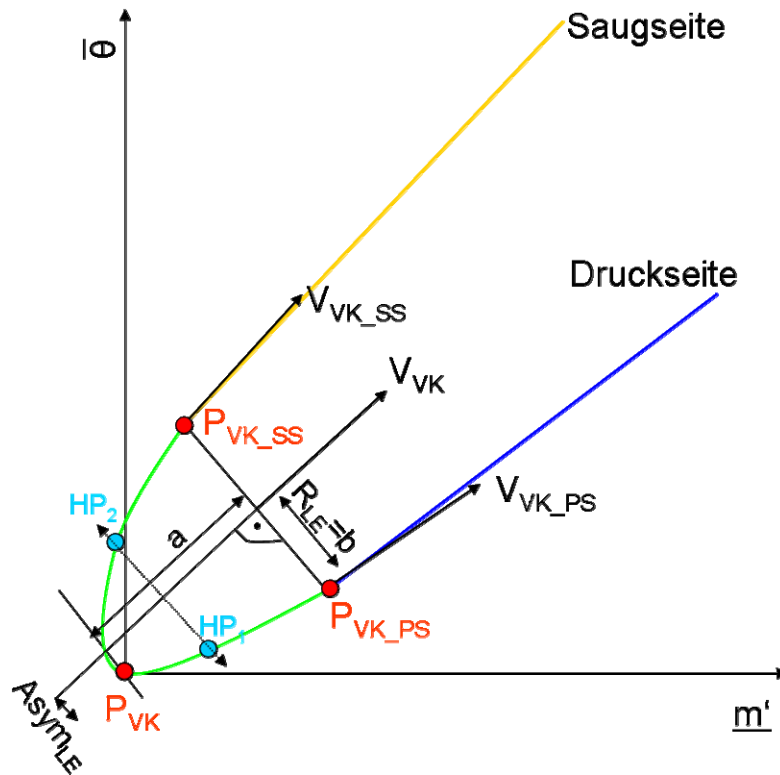


Bild 2.3: Designparameter und Profil-Konstruktion.

Somit stehen die Anfangs- und Endpunkte sowie die ersten Ableitungen an diesen Punkten für Saug- und Druckseite fest und diese Profil-Splines können gemäß der weiter unten beschriebenen Methode fertig konstruiert werden.

Die finale Gestalt der Vorderkante ergibt sich im Anschluss an die Konstruktion von Druck- und Saugseite indem man einen interpolierenden B-Spline durch die Punkte P_{VK_SS} , P_{VK} und P_{VK_PS} bestimmt der geometrisch (nicht zwingend parametrisch) stetig, differenzierbar und krümmungsstetig in Saug und Druckseite übergeht. Zusätzlich kann die Form der Vorderkante noch über den Formfaktor an Vorderkante LE_{FF} variiert werden. Dieser staucht bzw. streckt die zu interpolierenden Hilfspunkte HP_1 und HP_2 in Bild 2.3 welche initial auf 25% bzw. 75% Lauflänge der Vorderkante liegen. Die Verschiebung aus dieser initialen Lage ergibt sich aus:

$$HP_1 = HP_{1_init} - LE_{FF} \cdot (P_{VK_SS} - P_{VK_PS})$$

$$HP_2 = HP_{2_init} + LE_{FF} \cdot (P_{VK_SS} - P_{VK_PS})$$

Die Vorderkante ist dann durch den interpolierenden B-Spline 3.Grades durch die Punkte P_{VK_SS} , HP_1 , P_{VK} , HP_2 und P_{VK_PS} gegeben. Die Hinterkante wird entsprechend gebildet wobei jedoch wieder auf die Freiheit der asymmetrischen Gestaltung verzichtet wird.

Saug- und Druckseite:

Die noch benötigten Designparameter für die Saug und Druckseite eines jeden Konstruktionsprofils sind gegeben durch Kontrollpunkte für die jeweiligen B-Spline Kurven. Die Anzahl dieser Kontrollpunkte ist variabel gestaltet und kann zwischen 5 und 9 variiert werden. Der Anfangspunkt und Endpunkt sowie die ersten Ableitungen an diesen Punkten wurden bereits aus den geometrischen Daten abgeleitet (Für die Saugseite sind dies: P_{VK_SS} , P_{HK_SS} und V_{VK_SS} , V_{HK_SS}).

Durch die Vorgabe der Ableitungen ergibt sich für den zweiten bzw. vorletzten Kontrollpunkt, dass nur eine Variable, nämlich \underline{m}' frei wählbar ist (siehe [Piegl und Tiller 1997]). Im Folgenden wird die \underline{m}' mit X und die $\underline{\theta}$ Koordinate mit Y bezeichnet. Somit ergeben sich bei z.B. 5 Kontrollpunkten die weiteren freien Designparameter für die Kontrollpunkte der Saugseite wie folgt:

Kontrollpunkt 1: P_1	-	(Gegeben durch P_{VK_SS})
Kontrollpunkt 2: P_2	X_{2SS}	(2. Koordinate durch V_{VK_SS} gegeben)
Kontrollpunkt 3: P_3	X_{3SS}, Y_{3SS}	
Kontrollpunkt 4: P_4	X_{4SS}	(2. Koordinate durch V_{HK_SS} gegeben)
Kontrollpunkt 5: P_5	-	(Gegeben durch P_{HK_SS})

Die jeweiligen Koordinaten der Kontrollpunkte liegen dabei in einem normierten Koordinatensystem vor, in dem der Einheitsvektor (1,0) durch den Verbindungsvektor vom ersten zum letzten Kontrollpunkt gegeben ist.

Mit diesen Werten und der geeigneten Wahl eines Knotenvektors U und des Grades p, ist die Saugseite als B-Spline Kurve vollständig definiert:

$$C(u) = \sum_{i=0}^n N_{i,p}(u) \cdot \vec{P}_i, \quad u \in [a; b]$$

$$U = (u_0 = a, \dots, u_p = a, \quad u_{p+1}, \dots, u_{m-p+1}, \quad u_{m-p} = b, \dots, u_m = b)$$

Die Basisfunktionen N_{ip} dieser B-Spline Kurven sind z.B. in [Piegl und Tiller 1997] S.20 definiert.

Für die Parametrisierung der Druckseite stehen zwei Möglichkeiten zur Verfügung: Zum einen kann die Druckseite, genauso wie die Saugseite, über die Wahl von Kontrollpunkten einer B-Spline Kurve bestimmt werden. Dies sei hier noch einmal am Beispiel von 7 Kontrollpunkten verdeutlicht.

Parametrisierung1:

Kontrollpunkt 1: P_1	-	(Gegeben durch P_{HK_DS})
Kontrollpunkt 2: P_2	X_{2SS}	(2. Koordinate durch V_{HK_DS} gegeben)
Kontrollpunkt 3: P_3	X_{3SS}, Y_{3SS}	
Kontrollpunkt 4: P_4	X_{4SS}, Y_{4SS}	
Kontrollpunkt 5: P_5	X_{5SS}, Y_{5SS}	
Kontrollpunkt 6: P_6	X_{6SS}	(2. Koordinate durch V_{VK_DS} gegeben)
Kontrollpunkt 7: P_7	-	(Gegeben durch P_{VK_DS})

Diese Parametrisierung der Profile erfüllt alle Bedingungen die am Anfang dieses Kapitels an eine Geometrieparametrisierung von Verdichterschaufeln gestellt wurden. Sie hat jedoch den Nachteil, dass einige Profileigenschaften welche insbesondere strukturelle Relevanz haben (z.B. maximale Profildicke und Füllungsgrad) nicht direkt gesteuert, sondern nur indirekt beeinflusst werden können.

Innerhalb einer automatischen aerodynamischen Optimierung sollen solche strukturellen Nebenbedingungen aber oft beibehalten werden. Daher wurde innerhalb dieses Projektes eine alternative Druckseitenbeschreibung entwickelt, welche die Druckseite so an eine mittels Parametrisierung 1 gegebene Saugseite anpasst, dass die Profildicke und der Füllungsgrad sowie die Lage der maximalen Dicke direkt festgelegt werden können.

Die Grundidee dabei ist, dass die Gestaltung der aerodynamisch dominanten Saugseite alle gestalterischen Freiheiten behält, während die Druckseite so an die Saugseite angepasst wird, dass die rudimentären strukturellen Anforderungen erfüllt werden.

Parametrisierung2

Hier wird nicht die Druckseite selbst, sondern eine Dickenverteilung parametrisiert, welche dann an diskreten Stützstellen von der Saugseite abgetragen wird und somit die Druckseite indirekt festlegt. Die Parameter dieser variablen Druckseitenparametrisierung sind in Bild 2.4 beschrieben

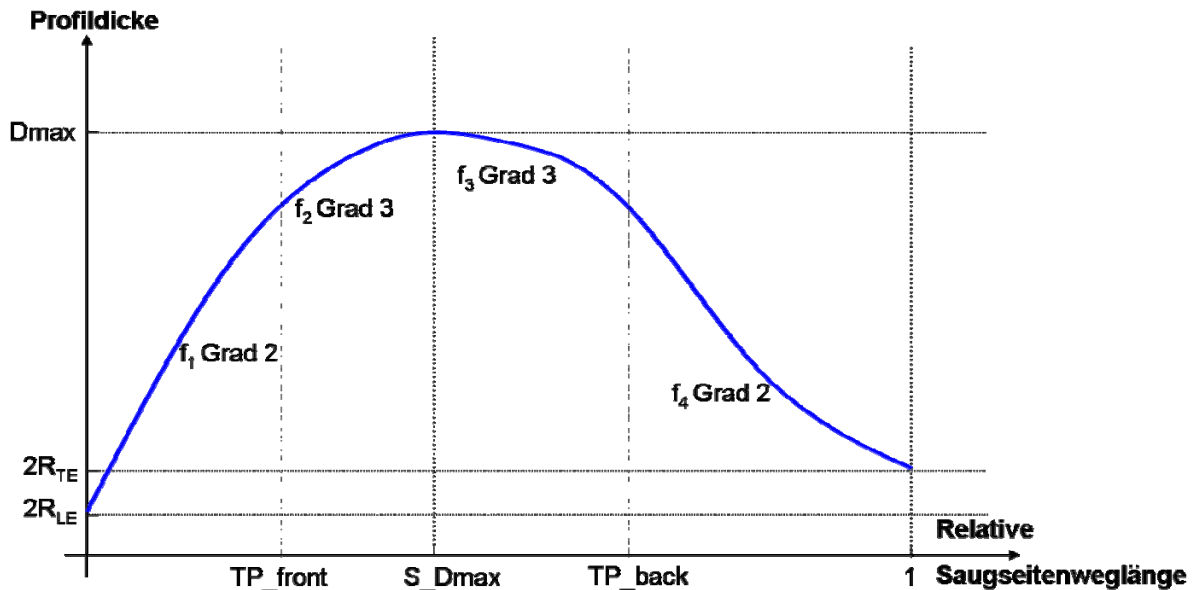


Bild 2.4: Designparameter der variablen Druckseite.

Die x-Achse stellt die relative Weglänge der bereits fertig konstruierten Saugseite von Punkt P_{VK_SS} an der Vorderkante zum entsprechenden Punkt P_{VK_SS} an der Hinterkante dar, die y-Achse stellt die Profildicke dar. Die blau eingetragene Polynomkurve, welche den Dickenverlauf beschreibt, ist aus 4 Polynomen f_i zusammengesetzt und ergibt sich aus den folgenden Parametern:

S_Dmax	Position der maximalen Profildicke
$Dmax$	maximale Profildicke
Fill	Füllungsgrad gesamt
FillFront	Füllungsgrad im Bereich von f_1 und f_2
TP_front	Übergangslage $f_1 \rightarrow f_2$
TP_back	Übergangslage $f_3 \rightarrow f_4$
dFrontArea	prozentuale Flächenvergrößerung im Bereich von f_1
dBackArea	prozentuale Flächenvergrößerung im Bereich von f_4

Insgesamt ergeben sich aus den Koeffizienten der 4 Polynome 14 Unbekannte (Beachte hierbei die unterschiedlichen Polynomgrade in Bild 2.4). Die dazugehörigen linearen Gleichungen mit denen diese bestimmt werden, lassen sich in zwei voneinander entkoppelte lineare Gleichungssysteme unterteilen. Für den vorderen Bereich ($S < S_D_max$) sollen die folgenden Bedingungen erfüllt werden:

1. $f_1(0) = 2 \cdot R_{LE}$
2. $f_1(TP_front) = f_2(TP_front)$
3. $f_1'(TP_front) = f_2'(TP_front)$
4. $f_2(S_D\max) = D\max$
5. $f_2'(S_D\max) = 0$
6.
$$\int_0^{TP_front} f_1 + \int_{TP_front}^{S_D\max} f_2 = FillFront \cdot D\max \cdot S_D\max$$
7.
$$\int_0^{TP_front} f_1 = A_0 \cdot \left(1 + \frac{dFrontArea}{100}\right)$$

In Gleichung 7 ist A_0 die Fläche unter f_1 , welche sich aus den Gleichungen 1. -6. ergibt wenn man f_2 in einem initialen Schritt ebenfalls als quadratisches (und nicht kubisches) Polynom ansetzt. Für die Polynome f_3 und f_4 ergeben sich entsprechende Gleichungen welche ebenfalls gelöst werden müssen.

Mit Hilfe dieser Polynome werden nun K diskrete Stützpunkte

$$P_{VK_PS} = Q_1, Q_2, Q_3, \dots, Q_K = P_{HK_PS}$$

senkrecht von der Saugseite an äquidistanten Weglängen abgetragen, so dass die gegebene Dickenverteilung an diesen Punkten erfüllt wird. Die Druckseite ergibt sich durch einen interpolierenden B-Spline 3. Grades durch diese Punkte.

Nachdem man sich für eine Druckseitenparametrisierung entschieden hat, liegen die Profile dann in Form von 4 B-Spline Kurven 3. Grades vor. Zur weiteren Behandlung werden diese 4 Kurven in Bezierabschnitte zerlegt und zu einer einzigen B-Spline Kurve zusammengefügt, wobei aus Stetigkeitsgründen darauf geachtet werden muss, dass keine echten Vielfachheiten innerer Knoten im Knotenvektor auftreten (siehe hierzu [Piegl und Tiller 1997]).

2.1.2 Fädelung

Nachdem die Profile im (m', θ) -Entwurfsraum konstruiert wurden und somit als 2-dimensionale geschlossene B-Spline Kurven vorliegen, müssen diese nun in den 3-dimensionalen Raum transformiert werden. Hierzu werden Informationen über die Konstruktionsstromlinie des jeweiligen Profils benötigt.

Jede Konstruktionsstromlinie muss dafür als Kurve in (x, r) -Koordinaten vorliegen (siehe Bild 2.5). Gemäß der in Bild 2.5 angegebenen Formel für m' gibt es auf dieser Stromlinie eine eindeutige Zuordnung:

$$\varphi : \mathbb{R}^2 \rightarrow \mathbb{R}$$

$$(x, r) \mapsto m'$$

Mittels der Umkehrabbildung kann man also die 2-dimensionalen Profilpunkte (m', θ) auf die Rotationsflächen der Konstruktionsstromlinien

(x, r, θ) bzw. auf $(x, y=r \sin(\theta), z=r \cos(\theta))$ abbilden, indem man x und r aus m' berechnet und θ beibehält.

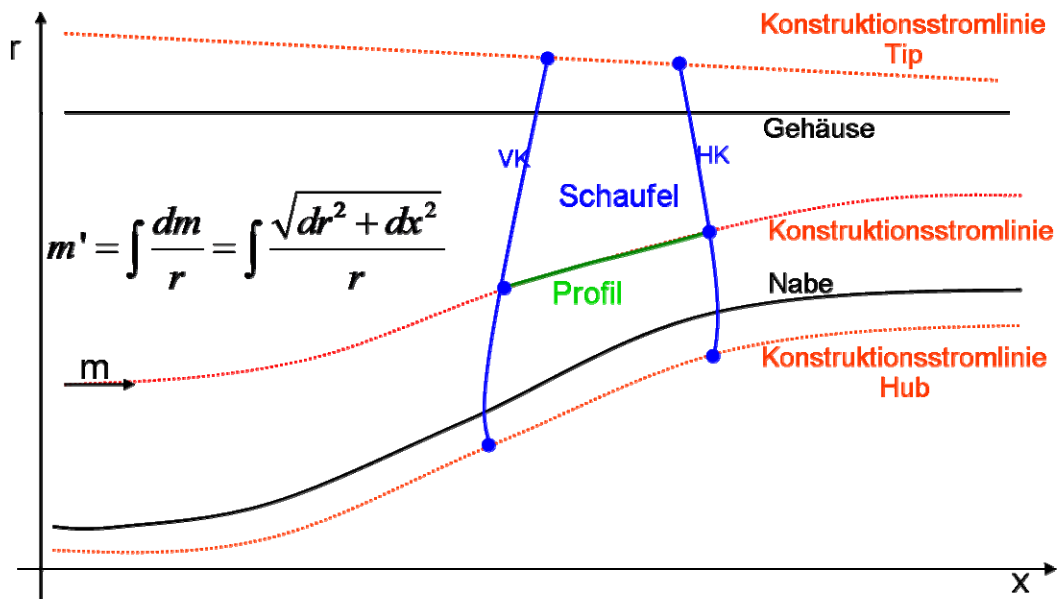


Bild 2.5: Verdichterschnitt und Stromlinien.

Da für die Profilierung im 2-dimensionalen Raum gilt, dass die Vorderkante bei $m'=0$ liegt, braucht man noch eine globale Integrationskonstante, welche im Bladegenerator als *GlobalShift* bezeichnet wird. Dieser *GlobalShift* kann wahlweise über die Maschinenachse x oder die Weglänge auf der Stromlinie m (z.B. für Radialverdichter) angegeben werden, und bewirkt eine Positionierung des untersten Konstruktionsprofils (Bezeichnung: Hubprofil) auf seiner Stromlinie. Die Verschiebung bzgl. θ wird so gewählt, dass der Flächenschwerpunkt des Hubprofils bei $\theta = 0$ liegt. Damit ist das Hubprofil positioniert.

Man kann nun, zur Positionierung der verbleibenden Konstruktionsprofile auf ihren Konstruktionsstromflächen zwischen 3 Typen der Fädelung wählen:

1. Schwerpunktfädelung

Die Profile werden so auf den Stromflächen verschoben, dass alle Flächenschwerpunkte gleiche $x-\theta$ Koordinaten haben

2. Vorderkantenfädelung

Die Profile werden so auf den Stromflächen verschoben, dass alle Vorderkantenpunkte gleiche $x-\theta$ Koordinaten haben

3. Hinterkantenfädelung

Die Profile werden so auf den Stromflächen verschoben, dass alle Hinterkantenpunkte gleiche $x-\theta$ Koordinaten haben

Nachdem nun alle Profile im 3-dimensionalen Raum auf ihren jeweiligen Stromflächen positioniert wurden und somit als 3-dimensionale B-Spline Kurven vorliegen, können sie noch aus dieser Ausgangsposition verschoben werden (ebenfalls auf den Stromflächen). Somit lassen sich z.B. bow, lean oder S-shape Schaufeln konstruieren.

Jedes Konstruktionsprofil kann also noch mit einem **x-shift**, **θ -shift**, **m-shift** oder **S-shift** (S-shift setzt sich aus x-shift und θ -shift zusammen, so dass die Verschiebung in Richtung der Sehnen des Profils verläuft) verschoben werden.

In einem letzten Schritt wird ein B-Spline Tensorproduktfläche gesucht, welche die Konstruktionsprofile enthält. Diese Prozedur wird als Surface-Skinning bezeichnet und ist beschrieben in [Piegl und Tiller 1997] S.457 f.f.

Diese Fläche wird dann im Step-Format ausgegeben und ist somit für alle nachfolgenden Programme in mathematisch eindeutiger Weise definiert und nutzbar. Die standardisierte Step-

Schnittstelle ermöglicht zusätzlich das verlustfreie Einlesen der Schaufel in alle gängigen CAD-Systeme.

Die oben beschriebene Methode zur Schaufelkonstruktion stellt die übliche Vorgehensweise dar. Gerade im Hinblick auf die automatische Optimierung wurden jedoch noch einige Varianten in den Bladegenerator implementiert. Bzgl. der Profilparametrisierung wurde mit der Parametrisierung² der Druckseite eine solche Variante auf Profilbasis vorgestellt. Die folgenden Varianten beziehen sich auf die 3D Gestaltung der Schaufel bei gegebener Profilierung.

Variante1 Fädeln nach Surface-Skinning

Hierdurch wird die Form der Fädellinie unabhängig von der Anzahl der Konstruktionsprofile. Man möchte z.B. mit nur wenigen Konstruktionsprofilen eine komplexe Vorderkantenform verwirklichen. Dafür werden die 2-dimensionalen Konstruktionsprofile wie oben beschrieben in den 3D Raum transformiert. Die Designparameter x-shift, θ -shift, m-shift und S-shift der einzelnen Konstruktionsprofile werden dabei jedoch gleich Null gesetzt.

Nach dem Surface-Skinning der Schaufel wird das entstandene Kontrollnetz der Schaufelfläche radial verfeinert ohne dabei die Schaufelform zu ändern. Danach wird eine Kurve der Form

$$C(u) = (h_{\text{rel}}(u), x_{\text{shift}}(u), m_{\text{shift}}(u), \theta_{\text{shift}}(u), S_{\text{shift}}(u))$$

eingelesen (Die neuen Designparameter sind disjunkte Punkte auf dieser Kurve aus denen sich dann die B-Spline Kurve errechnet). Diese Kurve gibt an, wie die jeweiligen Kontrollpunkte im Kontrollpunktnetz der Schaufelfläche (bei der jeweiligen relativen Schaufelhöhe h_{rel}) verschoben werden sollen.

Diese Vorgehensweise ermöglicht, gerade in der automatischen Optimierung, eine Reduktion der Anzahl von Designparametern, da mit der minimal notwendigen Anzahl von Konstruktionsprofilen gearbeitet werden kann ohne die Form der Fädellinie unnötig zu restringieren.

Variante2: Die Vorder- und Hinterkantenlinie werden vorgeschrieben.

Bei dieser Variante der Schaufelparmetrisierung wird der Verlauf der Vorder- und Hinterkantenlinie in (x,r) Koordinaten vorgeschrieben (siehe blaue Linien in Bild 2.5) und ergibt sich nicht durch das relativ komplexe Zusammenspiel von Fädeltyp (Schwerpunktfädelung, Vorderkantenfädelung, Hinterkantenfädelung) sowie x-shift, m-shift und S-shift.

Es werden für jede Konstruktionsstromlinie die Schnittpunkte mit der Vorderkante und Hinterkante in (x,r) Koordinaten berechnet (siehe blaue Punkte in Bild 2.5).

Die Differenz der m'-Werte dieser Punkt legt dann die axiale Länge des jeweiligen Konstruktionsprofils fest. Bei der Fädelung wird dann darauf geachtet, dass die Konstruktionsprofile zwischen diesen Punkten auf Ihren Stromflächen positioniert werden. x-shift, m-shift und S-shift werden gleich Null gesetzt und nur der θ -shift wirkt noch auf die einzelnen Profile.

Nach dem Surface-Skinning wird darauf geachtet, dass die Vorder- und Hinterkantenlinie der geskinnten Schaufel auch zwischen den Konstruktionsprofilen mit der Vorgabe übereinstimmen. Diese Anpassung geschieht wiederum durch Verschiebungen der Kontrollpunkte der Schaufel B-Spline Fläche.

Mit dieser Designvariante lassen sich z.B. sehr leicht Schaufeln entwerfen deren Axialspalt zu benachbarten Schaufeln nicht variiert, die Profilierung jedoch geändert werden soll.

Variante3: Cutting

Mit Cutting ist das Verschneiden einer 3D-Schaukel mit den rotationssymmetrischen Naben- und Gehäuseflächen des Verdichters gemeint. Hierbei sind innerhalb des Bladegenerators nur Interpolationsroutinen und keine Extrapolation erlaubt. Das bedeutet: Nabe- und Gehäusekurven in (x,r)-Koordinaten müssen innerhalb der Fläche liegen, welche durch die oberste Tippiprofil Konstruktionsstromlinie und die unterste Hubprofil Konstruktionsstromlinie eingeschlossen wird (siehe Bild 2.5). Falls keine Cutting-Option gewählt ist, wird davon ausgegangen, dass der

Nabenverlauf mit der Hubprofil Konstruktionsstromlinie und der Gehäuseverlauf mit der Tipprofil Konstruktionsstromlinie übereinstimmen.

Mittels dieser Bladegenerator Option wird für die Optimierung eine Variation des Strakverlaufs bei unveränderlicher Schaufelgeometrie ermöglicht.

Variante4: Radiale Verteilung der Profilparameter (PreBlade)

Für jeden Designparameter der Konstruktionsprofile wurde die Möglichkeit implementiert diesen, nicht für jedes Profil einzeln, sondern über einen radialen funktionalen Zusammenhang zu definieren. Dies wird mit einem eigenen Programm erreicht, welches dem Bladegenerator vorgeschaltet wird. Dieses Programm liest die radiale Verteilung der jeweiligen Designparameter ein und gibt die spezifischen Werte für die Konstruktionsprofile wieder aus. Dadurch können Designparameter eingespart werden, falls die Anzahl der Konstruktionsprofile größer als die zur Parametrisierung des radialen funktionalen Zusammenhangs benötigten Parameter.

Fazit: Der zum großen Teil innerhalb dieses Projektes entwickelte Bladegenerator des Instituts für Antriebstechnik, wurde nach Vorgaben entwickelt welche den Bedürfnissen der aerodynamischen Verdichter-Designer entsprechen:

- Nutzung herkömmlicher intuitiver Designparameter wie Staffelungswinkel, Keilwinkel, axiale Länge etc.
- Definierte Schnittstelle zu nachfolgenden Programmen über das Step-Format
- Möglichkeit zum Schaufelentwurf von Axial- und Radialverdichtern.
- Möglichkeiten zur lokalen und globalen Variation.

Zum anderen wurde der Bladegenerator in seinen Variationsmöglichkeiten an die Bedürfnisse der automatischen Optimierung angepasst:

- Die variable Druckseitenparametrisierung ermöglicht die *automatische* Sicherstellung von strukturmechanischen Nebenbedingungen in der Optimierung.
- Die Anzahl der Designparameter ist variabel. Durch die Variante1 *Fädeln nach Surface-Skinning*, die Variante4 *Radiale Verteilung der Profilparameter* oder die Wahl von nur wenigen Kontrollpunkten für die Parametrisierung der Profile, kann die Dimension des Suchraumes reduziert werden.
- Vorgabe von Vorder- und Hinterkantenverlauf ermöglicht z.B. die automatische Sicherung von Axialspalten bei der Optimierung mehrerer Schaufelreihen
- Cutting Option ermöglicht Strakvariation bei unveränderlicher Schaufelgeometrie.

Der Bladegenerator des Instituts für Antriebstechnik, und die hier beschriebene Form der Schaufelparametrisierung, haben sich bereits in vielen Projekten, sowohl bei der manuellen Auslegung von Axialverdichterstufen als auch im Rahmen der automatischen Optimierung, als sehr erfolgreich erwiesen.

2.2 Geometrische Restriktionen

Jede Verdichterkonfiguration wird, nachdem die Geometrie mittels Strakvariation und Bladegenerator erzeugt wurde, auf Ihre Sinnhaftigkeit überprüft. Je nach Wahl der Schaufelparametrisierung, der Wahl der zur Optimierung frei gegebenen Parameter und der zugelassenen Parametergrenzen, kann ein großer Prozentsatz der Verdichterkonfigurationen, welche ja mehr oder weniger zufällig erzeugt werden, als unzulässig eingestuft werden.

Solche unzulässigen Verdichter durchlaufen dann auch nicht den Rest der numerisch sehr kostspieligen Prozesskette sondern werden mit schlechten Zielfunktionswerten versehen sofort an den Optimierer zurück gesendet (wie in Bild 2.1 dargestellt).

Die Kriterien nach denen Verdichtergeometrien aussortiert werden orientieren sich am konstruktiven Verdichteraufbau und seien im Folgenden kurz beschrieben. Es werden

nacheinander alle Schaufelreihen der Verdichterkonfiguration durchlaufen. Sobald eine Schaufel als unsinnig eingestuft wird, wird der gesamte Verdichter verworfen.

Jede Schaufel wird wiederum zuerst auf die Plausibilität ihrer Konstruktionsprofile überprüft. Kriterien dabei sind:

- Die maximale Dicke D_{max}
- Lage der maximalen Dicke
- Füllungsrad
- Existenz lokaler Minima im Dickenverlauf
- Lage des Flächenschwerpunktes des Profils
- Lage der maximalen Wölbung
- Maximale Krümmung auf der Profilsaugseite
- Maximale Krümmung auf der Profildruckseite
- Profilfläche

Falls alle Kriterien für jedes Konstruktionsprofil erfüllt werden (die erlaubten Intervalle für obige Werte müssen in Input-Dateien des Optimierers editiert werden), werden noch einige Designparameter bzw. geometrische Eigenschaften der Schaufelprofile bzgl. ihrer radialen Verteilungen untersucht. Beispiele hierfür sind:

Staffelungswinkel	β_{ST}
Vorderkanten Metallwinkel	β_{LE}
Vorderkantenradius	R_{LE}
Hinterkanten Metallwinkel	β_{TE}
Hinterkantenradius	R_{TE}
Profilfläche	Area
Maximale Profildicke	D_{max}

Für diese Parameter werden die Kurven untersucht, welche sich aus der Abtragung des jeweiligen Parameters über der relativen Schaufelhöhe ergibt. Die Forderungen an diese Kurven können folgendermaßen aussehen:

- Monotonie-Forderung (fallend oder steigend)
- Anzahl lokaler Extrema

So sollten z.B. für einen Rotor die Parameter R_{LE} , R_{TE} , Area und D_{max} aus strukturellemechanischen Gründen monoton fallend über der Kanalhöhe sein.

Auf weitere vorstellbare Restriktionen, wie z.B. die Berücksichtigung von Krümmungsverteilungen oder Dickenverteilungen der 3D Schaufel wird hier verzichtet. Stattdessen wurde eine finite Elemente Strukturanalyse in die Prozesskette integriert auf die später in diesem Bericht eingegangen wird.

2.3 Netzerzeugung: Gridgenerator

Die **automatische 3-dimensionale** (nur auf diese soll hier eingegangen werden, Netzerzeugung für Mises oder Magellan wird in diesem Kapitel nicht besprochen) **Netzerzeugung** stellt bei variierenden Schaufelgeometrien nach wie vor eine große Herausforderung an strukturierte Netze dar. Bei herkömmlichen Netzgeneratoren führen größere Änderungen in der Schaufelgeometrie, bei nicht angepassten Netzparametern, oft zu Faltungen im Netz. Dies gilt sowohl für das CFD-Netz des Strömungslösers als auch für das FEM-Netz der Strukturrechnung.

Daher wurde innerhalb dieses Projektes ein eigenes Programm-Modul zur automatischen Netzerzeugung des FEM-Netzes und CFD-Netzes entwickelt dessen grober Ablauf im nachfolgenden Abschnitt kurz beschrieben wird.

Zu Beginn liest der Netzgenerator (im Folgenden als Gridgenerator oder AutoGrid bezeichnet) eine Schaufel im Step-Format ein. Danach werden das Hub- und Tipprofil mittels der (x,r)-

Koordinaten von Nabe und Gehäuse ins 2-dimensionale (m', θ) -Koordinatensystem transformiert (siehe Bild 2.5 bzw. die Funktion ϕ aus Kapitel 2.1.2).

Danach erfolgt die Vernetzung für CFD und FEM im 2-dimensionalen Koordinatensystem. Das CFD Netz liegt dann für Hub- und Tipprofil in einer O-C-H-H-H Topologie wie in Bild 2.6 dargestellt vor.

Das 2D-FEM Netz der Profile besteht aus einem H-Netz und 2 sog. Spezial-H grids für Vorder- und Hinterkante (Bild 2.7 zeigt das spezial H-grid für die Vorderkante und das einfache H-Netz).

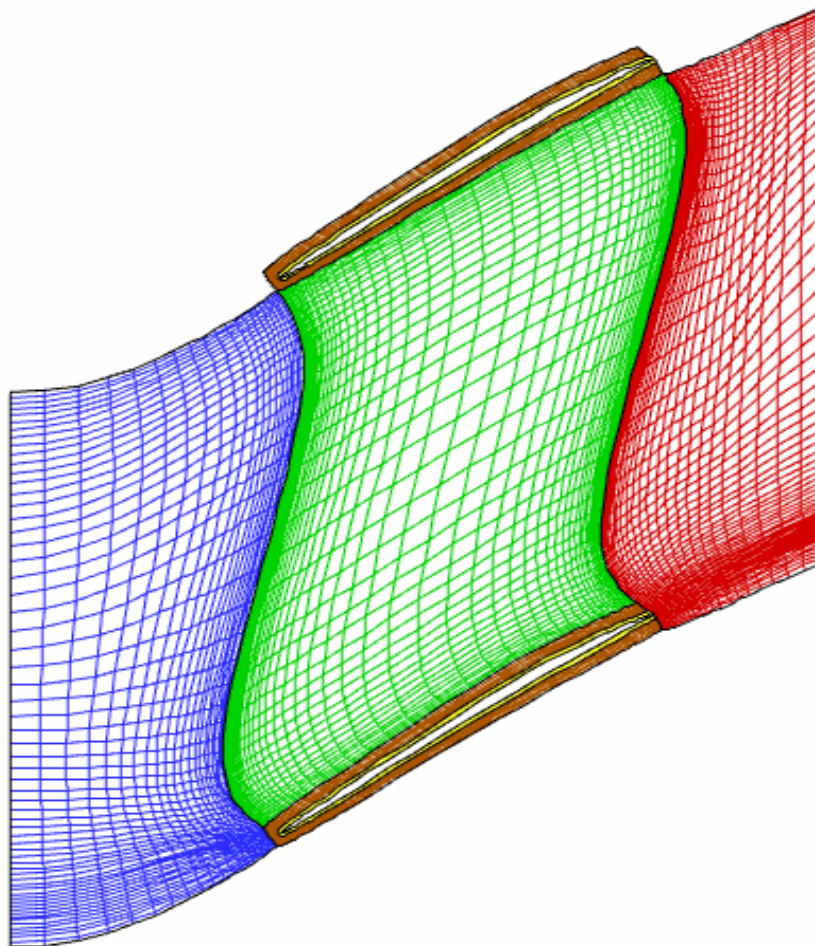


Bild 2.6 O-C-H-H-H Topologie des CFD-Netzes mittels AutoGrid

Diese strukturierten 2D-Netze werden dann in den 3-dimensionalen Raum zurück transformiert (gemäß der Umkehrabbildung von ϕ aus Kapitel 2.1.2). Danach werden die Schaufeloberflächennetze entworfen welches für das CFD und FEM unterschiedlichen Anforderungen genügen müssen (Das CFD Netz muss an Nabe und Gehäuse stärker verfeinert werden).

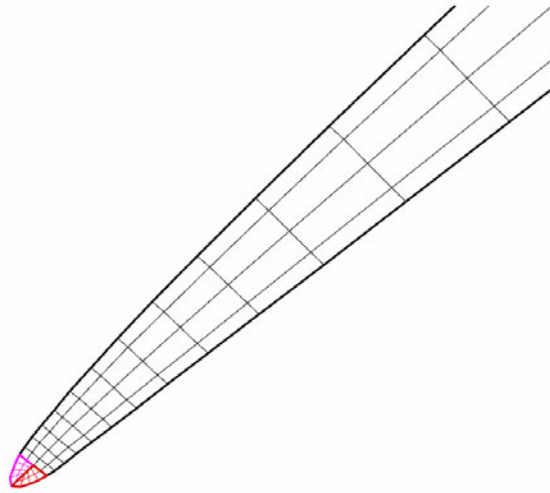


Bild 2.7 SpecialH-H-SpecialH Topologie des FEM -Netzes mittels AutoGrid

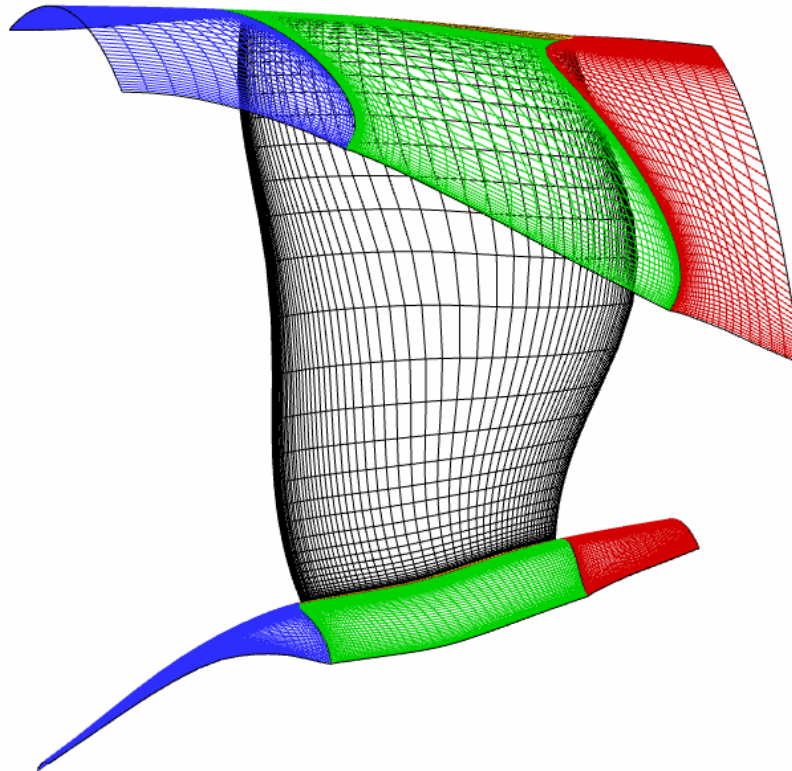


Bild 2.8 Hub-Grid, Surface-Grid und Tip-Grid nach AutoGrid

Nun liegen drei Netze auf 2-dimensionalen Flächen im 3-dimensionalen Raum vor (siehe Bild 2.8 für das CFD-Netz). Zur Fertigstellung der 3-dimensionalen Rechenetze muss der eingeschlossene Bereich noch gefüllt werden.

Zum Füllen des 3D CFD-Netzes wird dazu ein kommerzielles Programm (G3DMesh von CFD-Norway) genutzt welches auch das vom Strömungslöser benötigte cgns-Format bereit stellt und die Erzeugung von Spaltnetzen (Tip- und/oder Hub clearance) übernimmt.

Das Füllen des 3D FEM-Netzes erfolgt über eine transfinite Interpolation innerhalb des Gridgenerators, welcher auch die benötigten Output-Dateien für die FEM-solver Permas und Calculix erzeugt (siehe [Backhaus 2007]).

Die so erzeugten 3D Netze sind in Bild 2.9 und Bild 2.10 dargestellt.

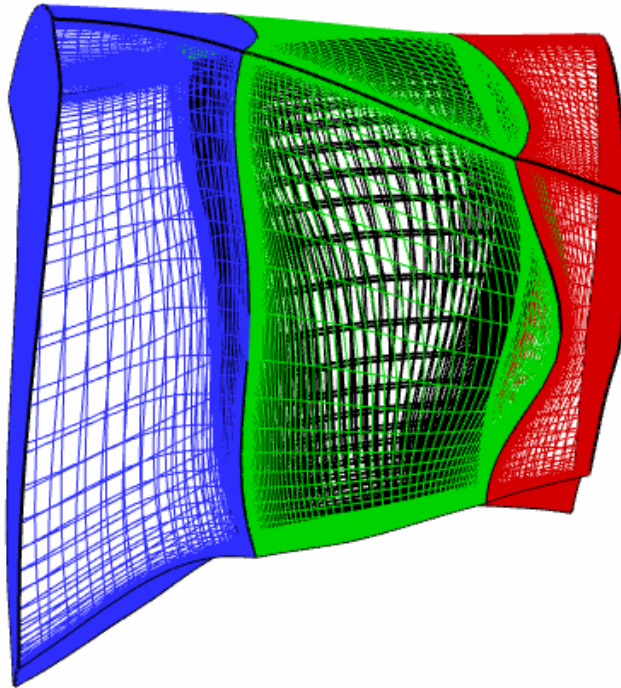


Bild 2.9 Finales 3D CFD-Netz

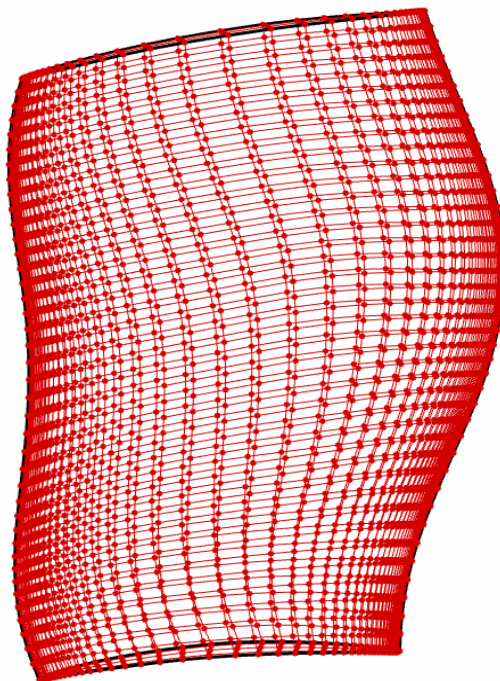


Bild 2.10 Finales 3D FEM-Netz

Der Vorteil der automatischen Netzerzeugung mittels AutoGrid liegt nicht in der Netzgüte sondern in der Stabilität mit der unterschiedliche Geometrien ohne manuelle Anpassung der Parameter vernetzt werden können. Während kommerzielle Programme zur Erzeugung blockstrukturierter Netze von Verdichtern oft hunderte von Netzparametern benötigen um qualitativ hochwertige Netze zu erzeugen, kommt *AutoOpti* mit einem Bruchteil (< 20) dieser Parameterzahl aus und erzeugt in sehr weiten Bereichen der Geometrieviation rechenbare Netze.

2.4 Strömungslösungen

In diesem Kapitel werden die unterschiedlichen Strömungslöser beschrieben die in die automatische Prozesskette des Optimierers *AutoOpti* integriert wurden. Sie unterscheiden sich sowohl durch Ihren Einsatzbereich innerhalb der Verdichterauslegung als auch durch den Umfang der von Ihnen benötigten numerischen Ressourcen.

Bild 2.11 zeigt den typischen Design-loop der Verdichterauslegung in dem 3 qualitativ unterschiedliche Strömungslöser vorkommen:

1. Ductflow oder Throughflowprogramme für die Verdichter-Vorauslegung
2. 2D oder Q3D blade to blade Löser für das aerodynamische Profildesign
3. 3D Navier Stokes Programme zur realitätsnahen detaillierten Strömungssimulation und 3D Gestaltung der Schaufeln

Jeder dieser Strömungslöser wird innerhalb des Design-Loops benötigt um möglichst optimale Geometrien und/oder Arbeitsverteilungen im jeweiligen Designschritt zu ermitteln. Daher lässt sich jedes dieser Programme mit einem Optimierer koppeln wobei der numerische Aufwand von Throughflow über blade to blade bis hin zu 3D Navier-Stokes sehr stark ansteigt.

Zusätzlich wurde innerhalb dieses Projektes ein gekoppelter Löser integriert auf den in Kapitel 2.4.3 eingegangen wird.

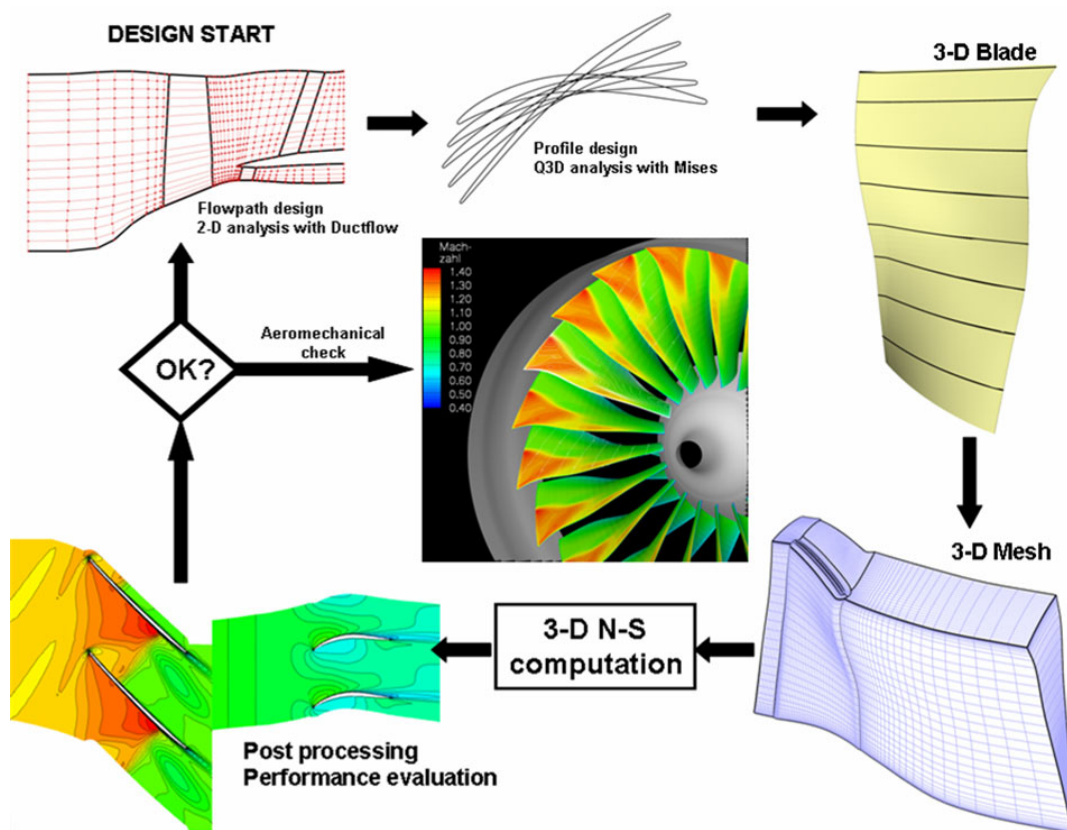


Bild 2.11: Design-Loop der Verdichterauslegung

Auf das aerodynamische Modell zur Verdichter-Vorauslegung, welches allgemein durch Meridianströmungsverfahren beschrieben wird, soll hier nicht näher eingegangen werden. Die Optimierung mittels Throughflow-Verfahren ist jedoch im COOREFF-T Vorhaben 1.1.2 (Förderkennzeichen 0327720A) der Universität Kassel [Becker/Lawerenz 2008] detailliert behandelt worden.

2.4.1 Q3D-Euler Grenzschicht MISES

Das Programm Mises ist ein Q3D gekoppeltes Euler-Grenzschichtverfahren zur blade to blade Strömungssimulation welches in [Drela 1998] detailliert beschrieben ist. Dieses Programmpaket ist unterteilt in die einzelnen Module

Iset	Netzgenerator
Ises	Strömungslöser
Iplot	Post-processing

und wird in der gesamten Turbomaschinenindustrie als validiertes Werkzeug zur aerodynamischen Profilauslegung genutzt.

Abgesehen von den physikalisch bedingten nicht konvergenten Simulationen im Programmteil Ises ist dieses Programm numerisch sehr stabil, und hat einen überschaubaren Speicher und CPU-Bedarf. Daher ist es zur Kopplung mit Optimierungstools wie *AutoOpti* hervorragend geeignet.

Nicht konvergente Simulationen können jedoch sehr viel Zeit benötigen bis Sie von selbst terminieren. Innerhalb einer Optimierungsstrategie, in der viele tausend solcher Simulationen durchgeführt werden, muss dieses Verhalten natürlich kontrolliert werden und nicht konvergente Simulationen frühzeitig unterbrochen werden. Daher ist für den Programmablauf von Mises ein eigenes kleines C-Programm (Bezeichnung CalcMises) entwickelt worden welches die Simulationen startet überwacht und gegebenenfalls abbricht. CalcMises wird für alle Betriebspunkte gestartet und benötigt den folgenden Input:

- Nummer des zu berechnenden Betriebspunktes
- Für jeden Betriebspunkt die Dateien:
 - Blade.Name
 - Stream.Name
 - Ises.Name
 - Gridpar.Name
 - Steuerungsdatei für Iset
 - Steuerungsdatei für Ises
 - Steuerungsdatei für Iplot
- Maximal zulässige Simulationszeit T_{\max}

Der Programmablauf von CalcMises ist dann sehr einfach. Zu Beginn des Programms wird ein zweiter Prozess mittels der C-Funktion fork() erzeugt.

Prozess2:

Dieser so genannte Kindsprozess hat die Aufgabe nacheinander die Programme Iset, Ises und Iplot mit den jeweiligen Inputdateien des zu rechnenden Betriebspunktes aufzurufen. Falls die Strömungssimulation konvergiert schreibt das post-processing Programm Iplot eine so genannte journal-Datei in der alle relevanten Strömungsgrößen, welche später für die Formulierung der Fitnessfunktion benötigt werden, zusammengefasst sind.

Nachdem diese Datei geschrieben ist terminiert sich der Kindsprozess selber.

Prozess1:

Dieser ursprüngliche Elternprozess hat lediglich die Aufgabe die maximal zulässige Simulationszeit T_{\max} abzuwarten und dann zu überprüfen ob der Kindsprozess die Simulation bis zu Ende durchführen konnte und ob diese konvergiert ist und somit eine gültige journal-Datei vorliegt.

Ist dies nicht der Fall terminiert der Elternprozess den Kindsprozess und alle Programme die vom Kindsprozess gestartet wurden (Iset, Ises und Iplot). Danach wird diese Information (es liegt keine

konvergente Lösung vor) an die automatische Prozesskette zurückgegeben und von dieser mit schlechten Zielfunktionswerten bestraft.

Ein ausführliches Beispiel für die Optimierung eines transsonischen Profilschnittes in mehreren Betriebspunkten mit Mises und *AutoOpti* findet sich in [Voß et al. 2006].

Da Mises nicht die Möglichkeit beinhaltet mehrere, aufeinander folgende Schaufelreihen bzw. Profilschnitte im Verbund zu simulieren, wurde innerhalb dieses Projektes die Möglichkeit implementiert, mehrstufige Profilschnitt-Optimierungen mit TRACE durchzuführen. Formal entspricht die Profilloptimierung mit TRACE der im folgenden Kapitel dargestellten 3-dimensionale Navier-Stokes Optimierung mittels TRACE. Daher wird auf diese Art der Profilloptimierung hier nicht näher eingegangen. Es müssen lediglich die folgenden Einschränkungen in der Prozesskette eingehalten werden:

Bladegenerator: Die Schaufeln werden nur mit einem Konstruktionsprofil (HubProfil = Tipprofil) prismatisch konstruiert.

Gridgenerator: In radialer Richtung wird das CFD-Netz mit nur 2 Netzebenen aufgelöst.

TRACE: Radial müssen als Randbedingungen reibungsfreie Eulerwände gesetzt werden.

2.4.2 3D Navier Stokes TRACE

Zur numerischen Berechnung von Strömungen in Turbomaschinen wird im DLR der am Institut für Antriebstechnik entwickelte Löser TRACE (Turbomachinery Research Aerodynamics Computational Environment) verwendet. TRACE ist ein Programmsystem zur stationären und instationären Berechnung von dreidimensionalen Strömungen in mehrstufigen Verdichter- und Turbinenkomponenten mit axialer und radialer Strömungsführung. Neben der Anwendung zur Untersuchung komplexer Strömungen am DLR und vielen weiteren Forschungs- und Universitätsinstituten, wird es seit einigen Jahren ebenfalls in der Industrie zur Entwicklung von Turbomaschinen, Triebwerken, Gasturbinen in Kraftwerken und ihren Komponenten eingesetzt. Merkmale dieses Lösert sind:

- Ein breites Einsatzspektrum
- Berechnung von stationären und instationären Strömungen in axialen und radialen Komponenten
- Nutzung strukturierter oder unstrukturierter Gitter
- Parallelisierte Rechnung einzelner Netzabschnitte.

Das Rechengitter wird hierzu in mehrere Blöcke unterteilt, die Strömungsinformationen über Interfaces austauschen und damit eine gleichzeitige Berechnung mehrerer Abschnitte und deutlich geringere Rechenzeiten erlauben.

TRACE basiert, wie ein Großteil der heute verwendeten Strömungslöser, auf der Lösung der Navier-Stokes Gleichungen. Diese werden anhand der Finite Volumen Methode diskretisiert und die Größen des erhaltenen Gleichungssystems anschließend mit Hilfe des RANS-Verfahrens in einen Mittel- und einen Schwankungswert aufgeteilt. Die dabei neu entstandenen Unbekannten, die als Reynoldsspannungen bezeichnet werden, erfordern die zusätzliche Verwendung eines Turbulenzmodells. Bei TRACE kann zwischen dem Eingleichungsmodell nach Spalart und Allmaras und dem $k-\omega$ Modell nach Wilcox, das zu den Zweigleichungsmodellen gehört, gewählt werden.

Neben dem während der Netzgenerierung erstellten Rechengitter benötigt TRACE weitere Vorgaben, die als Randbedingungen am Eintritt die radiale Verteilung von Totaldruck und -temperatur sowie des Anströmwinkels und der Turbulenzgrößen und am Austritt den statischen Druck vorsehen. Zur Vorgabe dieser Daten kann ein weiteres Programm gmcQT verwendet werden (alternativ können die sog. boundary-files auch direkt eingelesen werden), das ebenso die

Wahl der Rechenparameter, wie etwa der Anzahl der Iterationsschritte oder der verwendeten Methode zur Analyse der Grenzschicht, ermöglicht. Um ein schnelleres Konvergieren zu ermöglichen zusätzlich eine Initialisierung vorgenommen, in der einzelne Strömungsgrößen, wie z.B. Dichte oder Geschwindigkeit, mit Werten belegt werden. Diese Werte dienen der Rechnung dann als Initialisierung, aus der die tatsächlichen Größen ermittelt werden. Eine realitätsnahe Auswahl beschleunigt die Rechnung erheblich und kann z.B. aus bereits durchgeführten Rechnungen ähnlicher Betriebspunkte oder ähnlicher Geometrien bei identischer Netztopologie entnommen werden.

Da, wie bereits erwähnt, zur numerischen Simulation mit Hilfe des Navier-Stokes-Verfahren eine große Rechnerleistung gefordert ist, steht am Institut für Antriebstechnik ein eigenes Linux-Cluster zur Verfügung, das aus 190 Knoten mit je zwei 2GHz-Prozessoren besteht.

Ist die Rechnung schließlich konvergiert, werden die erhaltenen Daten mit Hilfe eines Post-Processing-Programms aufbereitet, so dass sie mit Hilfe von TECPLOT visualisiert und analysiert werden können. Weitere Informationen zu TRACE finden sich z.B. in [Nürnberger 2004], [Kügeler et al. 2001], [Weber 2004].

Die Kopplung von TRACE mit *AutoOpti* erfolgt in ähnlicher Weise wie die Einbindung von Mises. Da eine 3D TRACE-Rechnung erheblich mehr numerischen Aufwand bedeutet als eine Q3D Mises-Rechnung, ist auch das C-Programm (Bezeichnung: CalcTrace) welches die Simulationen startet überwacht und gegebenenfalls abbricht erheblich aufwendiger als das entsprechende Q3D-Überwachungsprogramm. Hinzukommt, dass eine TRACE-Rechnung aus mehreren Prozessen besteht die alle kontrolliert werden müssen und unterschiedliche Rechner mit lokalem Speicher nutzen. Im Folgenden wird die Wirkweise von CalcTrace kurz erläutert:

CalcTrace wird für alle Betriebspunkte gestartet und benötigt, ähnlich zu TRACE, die folgende Ordnerstruktur:

```
OperatingPointK/trace/  
OperatingPointK/trace/cgns/  
OperatingPointK/trace/cgns_restart/  
OperatingPointK/trace/cgns_start/  
OperatingPointK/trace/cgns_fs/  
OperatingPointK/trace/input/  
OperatingPointK/trace/residual/  
OperatingPointK/trace/run/  
OperatingPointK/trace/control/
```

Das ausführbare Programm CalcTrace.exe wird in OperatingPointK/CalcTrace.exe gestartet und benötigt als eigenen Input die Datei CalcTrace.input (beinhaltet Konvergenzkriterien) und die Datei SubClusterNodes aus der hervorgeht auf welchen Rechnern die parallelisierte TRACE-Rechnung stattfinden soll.

Nach dem Start erstellt das Programm CalcTrace zuerst einige ausführbare Shell-Skripte die im späteren Verlauf benötigt werden:

run.sh

Dieses Script startet TRACE (auf denjenigen CPU's die in der Datei SubClusterNodes angegeben sind)

get_and_clean.sh

Dieses Script sammelt nach der Trace-Rechnung von allen beteiligten Rechner-Knoten die Ergebnisse ein. Danach werden die Verzeichnisse auf diesen Knoten wieder für die nächste Rechnung gelöscht

copy_control.sh

Dieses Script sammelt während der Trace-Rechnung diejenigen Dateien von den Knoten ein aus denen der Konvergenzverlauf der Rechnung zu erkennen ist.

check_if_Trace_AliveOnNode.sh

Dieses Script kontrolliert ob auf jedem an der Simulation beteiligten Knoten innerhalb der letzten 900 Sekunden die jeweiligen cgns-files erneuert wurden. Ist dies nicht der Fall wird die Simulation als gescheitert angesehen.

kill_trace.sh

Dieses Script bricht die TRACE-Rechnungen auf allen beteiligten Knoten ab.

Danach wird, analog zu CalcMises, ein Kindsprozess erzeugt welcher die Strömungssimulation startet während der Elternprozess die Konvergenzkontrolle übernimmt.

Prozess2:

Der Kindsprozess startet die eigentliche Trace-Simulation über obiges Shell-Script (run.sh). Für den Fall, dass der Elternprozess Konvergenz feststellt werden: run.sh, TRACE und Kindsprozess vom Elternprozess (Prozess1 s.u.) terminiert. Ist dies nicht der Fall bis die maximale Anzahl von Trace-Iterationen gerechnet wurde, so beendet sich der Kindsprozess selber und hinterlegt eine Datei Failed_log. Diese soll dem Elternprozess signalisieren, dass TRACE unkonvergiert nach Erreichen der maximal zulässigen Iterationsschritte beendet wurde.

Prozess 1:

Der Elternprozess geht in eine Endlosschleife die er erst dann verlässt wenn einer der folgenden Fälle eintritt:

- Die Existenz der Datei Failed_log (s.o. Prozess2) zeigt an, dass TRACE unkonvergiert beendet wurde.
- Es wird Konvergenz detektiert (s.u.)
- Es wird Nicht-Konvergenz detektiert (s.u.)
- Das Skript check_if_Trace_AliveOnNode.sh detektiert eine gescheiterte Simulation.

Um die Konvergenz der Simulation zu überprüfen werden die sog. Residual-Dateien von allen Knoten eingesammelt (mittels copy_control.sh). Die im Folgenden benötigten und unten erläuterten Größen

N_{Min} , $N_{Residuum}$, $LowLimit_{Res}$, N_{mass} , $LowLimit_{Mass}$, $UpLimit_{Mass}$, $N_{efficiency}$, $LowLimit_{efficiency}$, $UpLimit_{efficiency}$, $N_{Divergenz}$, $LowLimit_{Konvergence_{Mass}}$
müssen der Datei CalcTrace.input entnommen werden.

Folgende Kriterien werden überprüft sobald eine gewisse minimale Anzahl von Zeitschritten N_{Min} durchlaufen wurde:

1. Residuum
Das durchschnittliche Residuum muss in den letzten $N_{Residuum}$ Zeitschritten dauerhaft unter einer Grenze $LowLimit_{Res}$ geblieben sein.
Der Gradient des Residuums muss im letzten Zeitschritt negativ sein.
2. Massenstrom
Der Ein- und Austrittsmassenstrom muss in den letzten N_{mass} Zeitschritten dauerhaft innerhalb eines Intervalls $[LowLimit_{Mass} , UpLimit_{Mass}]$ geblieben sein.
Der Gradient des Ein- und Austrittsmassenstrom muss im letzten Zeitschritt kleiner sein als im Durchschnitt der letzten N_{mass} Zeitschritte.
3. Wirkungsgrad

Der isentrope Wirkungsgrad muss in den letzten $N_{efficiency}$ Zeitschritten dauerhaft innerhalb eines Intervalls $[LowLimit_{efficiency}, UpLimit_{efficiency}]$ geblieben sein.

Konvergenz liegt vor wenn alle Bedingungen erfüllt sind. Ansonsten wird der Prozess für kurze Zeit angehalten und die Kontrollschleife startet erneut.

Nicht-Konvergenz wird folgendermaßen festgestellt

4. Nicht-Konvergenz

Falls nach $N_{Divergenz}$ Zeitschritten eine untere Grenze des Massenstromes $LowLimitKonvergence_{Mass}$ an allen Interfaces des Rechengebietes unterschritten ist, und die Massenstromgradienten an allen Interfaces negativ sind, wird davon ausgegangen, dass die Simulation nicht mehr konvergieren wird.

Nachdem Prozess1 die Kontrollschleife verlassen hat steht fest ob die Simulation erfolgreich war oder nicht.

Es werden die Simulationsergebnisse eingesammelt (get_and_clean.sh), alle Prozesse beendet (kill_trace.sh) und die Konvergenzinformation an die automatische Prozesskette zurückgegeben.

Ein ausführliches Beispiel für die Optimierung eines Axialverdichters mit TRACE und *AutoOpti* findet sich in [Dorfner et al. 2007].

2.4.3 Gekoppelter Löser TRACE-MAGELAN

Während für Throughflow, Q3D-Euler, und 3D-Navier-Stokes Optimierungen die Hauptaufgaben innerhalb dieses Projektes in der Automatisierung und der Kopplung mit *AutoOpti* lagen, musste hier zusätzlich die Kopplung der Löser untereinander entwickelt und die Konvergenz des gekoppelten Löser anhand von Beispielsimulationen getestet werden. Die Motivation für einen solchen gekoppelten Löser ist im Folgenden kurz erläutert.

In praxisnahen aerodynamischen Optimierungen ist die Gesamtrechnungszeit ein wesentlicher Faktor da die 3D NS-Simulation numerisch sehr aufwendig ist. Der iterative Entwurf eines mehrstufigen Verdichters ist daher zurzeit auf der Basis dreidimensionaler Navier-Stokes Verfahren nur eingeschränkt möglich. Vor diesem Hintergrund wird in den hier durchgeführten Untersuchungen die dreidimensionale Rechnung auf wenige Schaufelreihen beschränkt und somit gleichzeitig die Zahl der Entwurfparameter auf ein akzeptables Maß reduziert.

Die dreidimensionale NS-Rechnung wird, um das aerodynamische Verhalten des Gesamtverdichters in der Zielfunktionsformulierung berücksichtigen zu können, in eine Meridianströmungsrechnung für den Gesamtverdichter integriert. Diese Throughflow-Rechnung liefert nicht nur Informationen über die Veränderungen des Gesamtverdichterverhaltens bei Variation der 3D simulierten Stufen, sondern ermöglicht es zugleich, die Ein- und Austrittsbedingungen der 3D-Rechnung an die verschiedenen Geometrien der Optimierung anzupassen.

Wenn die beiden Strömungslöser gekoppelt werden, müssen sie Informationen untereinander austauschen. Bild 2.12 zeigt diesen Informationsaustausch zwischen den Programmen TRACE und MAGELAN und ist dem Abschlussbericht (Förderkennzeichen 0327720A) der Universität Kassel entnommen [Becker/Lawerenz2008]. Der Programmablauf des resultierenden gekoppelten Löser ist im Folgenden kurz skizziert:

Der gekoppelte Strömungslöser startet mit einer Throughflow-Rechnung (Throughflow-programm: MAGELAN) für den Gesamtverdichter, in der für alle Gitter die internen Korrelationen (siehe [Becker/Lawerenz2008]) benutzt werden. Da im Vergleich zu 3D-Rechnungen diese sehr schnell ablaufen, lässt man die Throughflow Rechnungen auskonvergieren. Falls diese Anfangsrechnung erfolgreich ist, werden die 3D-Randbedingungen (Strömungswinkel,

Totaldruck, Totaltemperatur am Eintritt sowie Abströmwinkel und Gitterverluste am Austritt), wie in Bild 2.12 skizziert, extrahiert und an die 3D-Rechnung übergeben. Mit diesen Randbedingungen wird die 3D-Simulation mittels CalcTrace und TRACE gestartet. Nach einer festgestellten Pseudo-Konvergenz (grobe Konvergenzeinstellung in CalcTrace) wird die Rechnung angehalten und aus den Daten werden durch Umfangsmittelung die Informationen über Verluste und Abströmwinkel berechnet. Es schließt sich eine weitere Throughflow-Rechnung an. Anstelle der Korrelationen werden jetzt für den 3D gerechneten Verdichterteil die Daten aus der 3D-Rechnung (Abströmwinkel und Verluste) verwendet. Für die restlichen, nicht dreidimensional berechneten Gitter werden weiterhin die Korrelationen benutzt. Ist der Durchlauf erfolgreich, werden die Randbedingungen der 3D-Rechnung wieder aktualisiert und das dreidimensionale Navier-Stokes Verfahren fortgesetzt. Dieser Zyklus wiederholt sich so lange, bis die 3D-Rechnung konvergiert ist (bzgl. enger gefasster Konvergenzkriterien) und die 3D-Randbedingungen sich nicht mehr verändern. In diesem Fall ist der gekoppelte Löser konvergiert. Falls jedoch ein Schritt in diesem iterativen Prozess fehlschlägt, gilt die gekoppelte Rechnung als nicht konvergiert.

Die Kopplung der Strömungslöser untereinander und die Kopplung mit *AutoOpti* erfolgten in enger Zusammenarbeit mit der Universität Kassel. Innerhalb dieser Kooperation der Forschungsinstitute sind alle notwendigen Anpassungsarbeiten durchgeführt worden (für Details hierzu siehe [Becker/Lawrenz2008]).

Die erfolgreiche Evaluation dieses neuen aerodynamischen Modells, gekoppelt mit *AutoOpti*, erfolgte am Beispiel des IDAC3-Verdichters der RWTH Aachen. Die Entwicklung des gekoppelten Löser und die aerodynamische Optimierung des IDAC3 wurden auf dem Linux Cluster des Instituts für Antriebstechnik des DLR durchgeführt. Die Ergebnisse der Optimierung wurden in einer gemeinsamen Veröffentlichung des DLR und der Universität Kassel publiziert [Becker et. al. 2008].

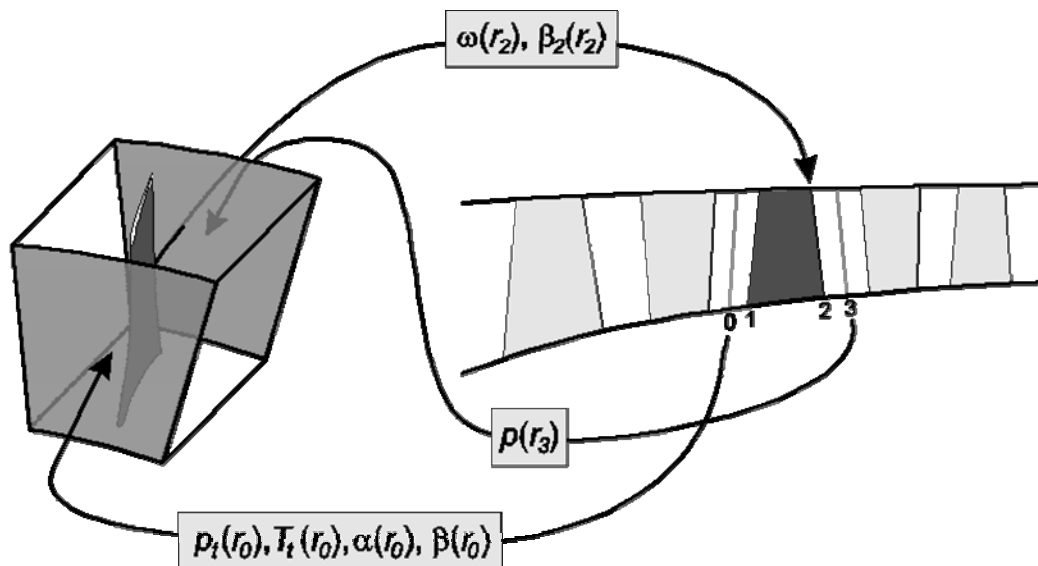


Bild 2.12 Kopplung MAGELAN-TRACE

2.5 FEM-Strukturanalyse

Die Einbindung der FEM-Strukturanalyse in die automatische Prozesskette ist ein wesentlicher Baustein von der rein aerodynamischen Optimierung hin zur multidisziplinären Optimierung. Als FEM-Programm können wahlweise die Programme PERMAS oder Calculix genutzt werden. Die Arbeiten zur Einbindung dieser Module fanden teilweise innerhalb des hier beschriebenen Projektes statt, und seien im Folgenden kurz beschrieben.

Bild 2.13 (entnommen [Backhaus 2007 b]) zeigt die Einbindung des FE-Lösers in die automatische Prozesskette von *AutoOpti*. Der Netzgenerator (siehe Kapitel 2.3) wurde dafür dahingehend verifiziert die FE-Netze sowohl im Permas als auch im Calculix kompatiblen Format zu erzeugen

Man erkennt aus Bild 2.13, dass die Strukturanalyse zum einen als Restriktion wirken kann: Vom Optimierer erzeugte Member welche gewissen Stabilitätsanalysen nicht genügen, werden nicht mehr CFD bewertet. Somit kann die FE-Analyse den gesamten Optimierungsprozess beschleunigen, da sie die Anzahl der numerisch kostspieligeren CFD-Berechnungen reduziert. Zum anderen können die Ergebnisse der FE-Analyse für jeden Member abgespeichert werden und in die Zielfunktionen einfließen.

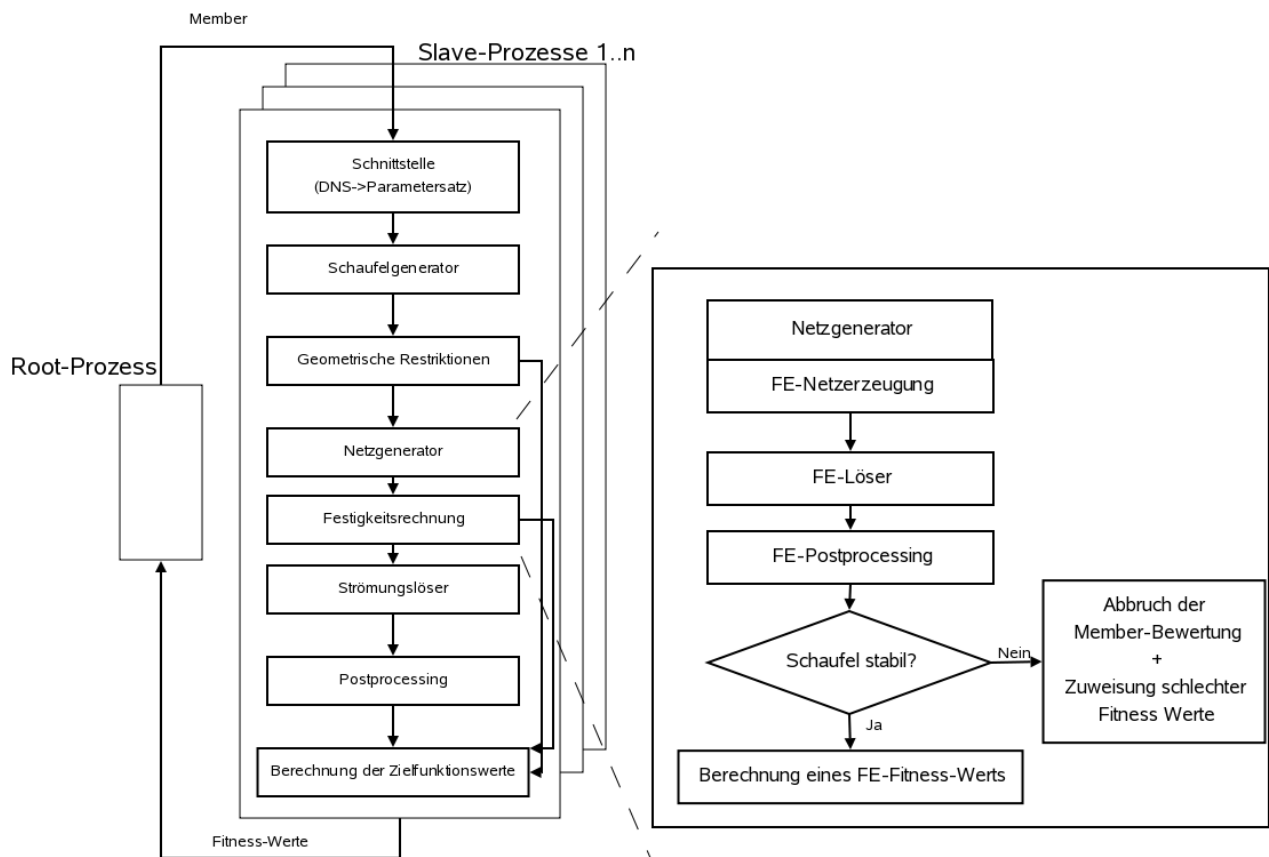


Bild 2.13 Einbindung des Finite Element Löser in die automatische Prozesskette

Der FE-Löser kann dabei prinzipiell auf zwei Weisen genutzt werden:

1. Statische Strukturrechnung → Mises Spannungen
2. Dynamischen Eigenfrequenzberechnung → Campbell Diagramm

Als **statisch** fest gilt eine Verdichterschaufel, wenn die aufgebrachte Belastung (Zentrifugalkräfte und Druckkräfte) nur linear elastische Verformungen bewirkt, also kein Fließen des Materials stattfindet. Zur Bewertung werden dabei die durch die statische Finite-Elemente Simulation berechneten resultierenden Spannungen herangezogen. Diese können entweder nach von Mises oder nach Tresca berechnet werden. Die von Mises-Vergleichsspannungen sind dabei die häufiger verwendeten. Grundsätzlich darf an keiner Stelle der Verdichterschaufel eine Vergleichsspannung auftreten, die größer ist als die Streckgrenze des Materials. Zusätzlich muss noch eine Sicherheitsreserve zur Streckgrenze eingehalten werden. Hierbei werden Erfahrungswerte verwendet. Für eine häufig verwendete Titanlegierung darf die Vergleichsspannung in einer Schaufel beispielsweise nicht über 500 Megapascal betragen, im Bereich der Schaufelvorderkante sind nur 300 Megapascal akzeptabel.

Zur Einbindung der statischen Festigkeitsrechnungen in die Prozesskette von *AutoOpti* siehe [Backhaus 2007 b].

Eigenfrequenzen bezeichnen die Frequenzen bei denen die größten Schwingungsamplituden eines Festkörpers auftreten.

Um die Ergebnisse der dynamischen Eigenfrequenzberechnung anschaulich darzustellen, verwendet man ein Campbell Diagramm, welches im Folgenden näher erläutert werden soll. Mittels dieses Diagramms lassen sich Kriterien überprüfen welche dem Zweck dienen Grenzen und Normen zu definieren, welche eingehalten werden müssen, um ein Versagen der Triebwerksschaufeln durch dynamische Probleme so weit wie möglich ausschließen zu können.

Das Campbell Diagramm (siehe Bild 2.14) stellt die verschiedenen Eigenfrequenzen und Anregungen dar, die Abszisse beschreibt hierbei die Drehzahl der Welle und ist somit ein Maß für die entstehende Fliehkraftbelastung. Die Ordinate beschreibt die Eigenfrequenz in Hz. Die zu den Eigenfrequenzen gehörenden Eigenmoden werden mit 2 Zeichen bezeichnet. Das erste Zeichen gibt die Nr. des vorhandenen Modentyps an und das zweite Zeichen gibt den Modentyp selbst an. Z.B. 1F = erste Biegeschwingung (F für Flexion), 1T = erste Torsionsschwingung, 2F = zweite Biegeschwingung usw.

Die Geraden, welche mit EO bezeichnet werden, stellen die Anregungen dar, wobei die erste Engine Order 1 Anregung pro Umdrehung bedeutet, die zweite Engine Order zwei Anregungen pro Umdrehung usw. Mit der Drehzahl lässt sich so die Frequenz der jeweiligen Engine Order ausrechnen. Es werden grundsätzlich immer die ersten 5 Engine Orders beachtet, da durch Unregelmäßigkeiten der Welle oder der Schaufeln, Anregungen bei niedrigen Frequenzen auftreten können. Zudem werden alle Anregungen berücksichtigt, welche durch Schaufelreihen verursacht werden die vor oder hinter der zu berechnenden Schaufel liegen. Dies sind bei den meisten Triebwerken Statoren, welche hinter oder vor einer Rotorreihe liegen. Wenn pro Umdrehung ein Rotor an angenommen 63 Statoren entlangläuft, wird dieser

Rotor 63-mal pro Umdrehung durch das entstehende Druckfeld angeregt. Diese Anregung wird ebenfalls als Engine Order EO63 eingezeichnet

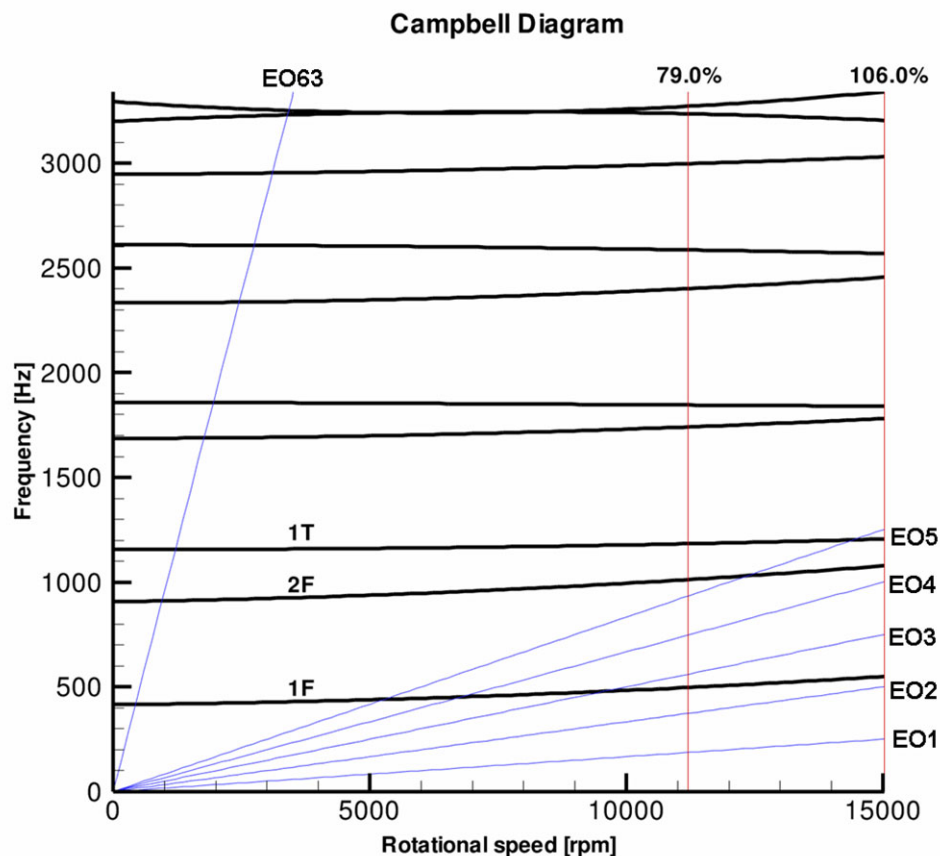


Bild 2.14 Beispiel eines Campbell-Diagramm einer hoch belasteten Rotorscheufel

Folgende Kriterien bzgl. des Campbelldiagramms sollten in der Schaufelauslegung beachtet werden:

- minimaler Abstand von 10% der Eigenmoden von den Anregungsgeraden bei Red Line bezogen auf die Eigenfrequenz des jeweiligen Eigenmodes
- minimaler Abstand von 10% zwischen den ersten Biege- und Torsionsmoden (1F und 1T bzw. , 2F und 1T) im gesamten Betriebsbereich.
- So genannte Crossings (Schnittpunkte zwischen EO und Moden) sollen im gesamten Betriebsbereich vermieden werden.

Zur Einbindung des dynamischen Strukturverhaltens mittels des Campbell-Diagramms in die Prozesskette von *AutoOpti* siehe [Schmitz 2008]

2.6 Parameter/Zielfunktionen

In diesem Modul werden die Ergebnisdateien eingelesen welche vorher von den post-processing Programmen (CFD und/oder FEM) geschrieben wurden. Bei den aerodynamischen und/oder strukturmechanischen Optimierungen (siehe Bild 2.1) beinhalten diese üblicherweise die aerodynamischen radial und/oder umfangsgemittelten Größen sowie die statischen und/oder dynamischen strukturmechanischen Eigenschaften des Verdichters.

Typische aerodynamische Parameter welche für jeden relevanten Betriebspunkt und alle Schaufelreihen bzw. Stufen abgespeichert werden sollten sind:

Massenstrom : \dot{m} ; Totaldruckverhältnis : Π_{tot} ; statisches Druckverhältnis : Π_{stat} ;

Totaltemperatur : T_{tot} ; Wirkungsgrad : η_{is} ; Abtrömwinkel : α_{exit} ; Totaldruckverlust : ω

Typische strukturmechanische Parameter sind maximale und durchschnittliche Mises Spannungen sowie Eigenfrequenzen der betrachteten Schaufelblätter bei den relevanten Drehzahlen (siehe Campbell Diagramm: Kapitel 2.5).

AutoOpti unterscheidet grundsätzlich zwischen der Struktur *flowparam* und der Struktur *fitness*. Innerhalb der Struktur *flowparam* werden alle physikalischen Parameter abgespeichert die für einen Member von Interesse sind. Diese werden dann zum einen dazu verwendet die Zielfunktionen zu berechnen welche in der Struktur *fitness* abgespeichert werden. Zum anderen werden die physikalischen Parameter (*flowparam*) dazu genutzt Nebenbedingungen zu überprüfen und ggf. restart-Optionen zu ermöglichen (siehe Kapitel 3.1)

Da die Zielfunktionen sowie die interessierenden physikalischen Größen in jeder Optimierung neu definiert werden müssen, wurden innerhalb von *AutoOpti* zwei C-Funktionen bereit gestellt die vor jeder Optimierung vom Nutzer angepasst werden müssen:

CalculateFlowparam(flowparam): Belegung der Struktur *flowparam*

CalculateFlowparam2Fitness(flowparam,fitness): Belegung der Struktur *fitness*

3. Das Optimierungspaket *AutoOpti*

Bei *AutoOpti* handelt es sich um ein auf der Evolutionsstrategie basierendes, mittels MPI parallelisiertes und durch Metamodelle beschleunigtes Programmpaket zur Mehr-Ziel-Optimierung (Multi Objective) beliebiger Zielfunktionen unter Einhaltung von Nebenbedingungen. Bei der Entwicklung von *AutoOpti* wurde, im Hinblick auf die aerodynamische Optimierung, besonderer Wert darauf gelegt, numerisch kostspielige Zielfunktionen (deren Gradienten i.A. unbekannt sind) die von vielen freien Parametern abhängen optimieren zu können.

Bei der aerodynamischen/strukturmechanischen Verdichterauslegung müssen teilweise konträre Ziele verfolgt und Kompromisse gefunden werden. Wenn diese unterschiedlichen Forderungen in einer Einzeloptimierung (SO Single-Objective) berücksichtigt werden sollen, müssen die einzelnen Kriterien gegeneinander gewichtet und zu einer einzigen Zielfunktion zusammengefasst werden. Die Gewichtung muss folglich vor dem Start der Optimierung erfolgen und legt die Suchrichtung im Zielfunktionsraum fest.

Der Ansatz der Mehr-Ziel-Optimierung (MO) bietet die Möglichkeit, mehrere Ziele gleichzeitig zu verfolgen. Nach dem Abschluss der Optimierung muss der/die verantwortlichen Designer aus der Menge aller optimalen Lösungen eine Konfiguration als Kompromiss auswählen. Evolutionäre Algorithmen gelten als besonders geeignet für MO-Optimierungen, da sie im Verlauf der Optimierung viele Parametersätze erzeugen und somit eine große Datenbasis für die Bestimmung der Paretofront liefern. Die Begriffe Dominanz, Paretorang und Paretofront seien im Folgenden kurz definiert

Definition 3.0

Ein Tupel freier Parameter (x_1, \dots, x_n) eines Optimierungsproblems mit n -dimensionalen Parameterraum und k -dimensionalen Zielfunktionsraum, bezeichnet man als **Member** oder Individuum. Sobald diesem Member seine Zielfunktionswerte y_k zugeordnet wurden bezeichnet man den Vektor $(x_1, \dots, x_n, y_1, \dots, y_k)$ als bewerteten Member.

Definition 3.1 (siehe Bild 3.1)

f_1, \dots, f_k seien die k Zielfunktionen einer multi-objective Optimierung. Man sagt, dass ein bewerteter Member M_1 einen Member M_2 **dominiert**, falls dies bzgl. aller Zielfunktionen gilt:

$$M_1 \prec M_2 \Leftrightarrow f_i(M_1) \leq f_i(M_2) \forall i \in \{1, \dots, k\} \wedge \exists j \in \{1, \dots, k\} : f_j(M_1) < f_j(M_2)$$

Definition 3.2

Der **Paretorang** eines Members ist definiert als:

$$P(M) := \#\{\text{bewertete Member } \hat{M} \mid \hat{M} \prec M\} + 1$$

Definition 3.2 (siehe rote Member in Bild 3.1)

Die **Paretofront** einer Optimierung besteht aus allen Members mit Paretorang 1:

$$PF := \{\text{bewertete Member } \hat{M} \mid P(\hat{M}) = 1\}$$

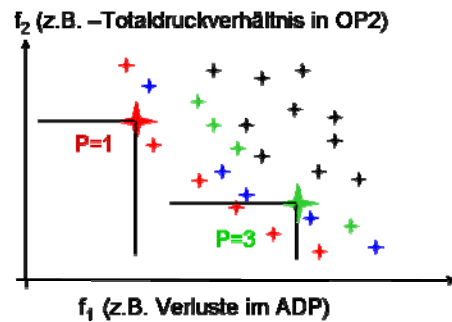


Bild 3.1: Definition Paretorang am Bsp. zweier Zielfunktionen: rot=Rang1; blau=Rang2; grün=Rang3;

3.1 Aufbau und Parallelisierung

Der prinzipielle Aufbau von *AutoOpti* lässt sich am Besten an Hand des Flussdiagramms in Bild 3.2 erläutern.

AutoOpti ist mittels MPI parallelisiert. Wie bereits in Kapitel 1.1 dieses Berichtes beschrieben, liegt die Aufgabe der slave-Prozesse in der Zielfunktionsauswertung und somit im Durchlaufen der automatisierten Prozesskette, welche im letzten Kapitel detailliert beschrieben wurde. Der Nutzer von *AutoOpti* muss daher zu Beginn jeder Optimierung die Anzahl der Slave-Prozesse und ggf. die Anzahl der pro Slave-Prozess benötigten CPU's angeben (TRACE Optimierung siehe Kapitel 2.4.2 und 2.4.3).

Der Root-Prozess (in Bild 3.2 grau hinterlegt) beinhaltet die eigentliche Optimierung. Er empfängt für jeden versendeten Satz freier Parameter der Optimierung vom Slave-Prozess (in Bild 3.2 orange hinterlegt) die Zielfunktionswerte (fitness) und physikalischen Größen (flowparam) und somit einen bewerteten Member. Danach wird, im Abgleich mit der Datenbasis aller bewerteter Member, der Paretorang dieses neuen Members bestimmt und er wird mittels dieses Gütekriteriums in die Datenbasis einsortiert und abgespeichert. Abgespeichert werden alle physikalischen Parameter (flowparam), alle Zielfunktionswerte (fitness), der jeweilige Vererbungsoperator (s.u.), der Paretorang, und die freien Parameter des Members. Im nächsten Schritt werden einige Member aus der Datenbank ausgewählt die Eltern für ein neues Individuum werden können. Bei der Elternauswahl sind zwei Kriterien zu beachten:

- 1 Je kleiner der Paretorang eines Members desto größer die Wahrscheinlichkeit des Members als Elternteil selektiert zu werden
- 2 Keine Region der Paretofront sollte für die Elternauswahl bevorzugt werden (wenn dies nicht über spezielle Input-Parameter eingestellt wird)

Nun wird mittels einer Zufallszahl ein Vererbungsoperator gewählt mit dem ein neuer Member erzeugt werden soll. Hierbei kann zwischen vier Operatoren gewählt werden die mit unterschiedlichen Wahrscheinlichkeiten (gesteuert über Input-Parameter) für die Neuerzeugung eines Members herangezogen werden können:

- Differential Evolution ([Becker/Lawerenz 2008])
- Mutation ([Ahmed 2005])
- Crossover ([Ahmed 2005])
- Gradient-Shot ([Becker/Lawerenz 2008])

Der neu erzeugte Member wird dann wieder zur Zielfunktionsauswertung an den gleichen Slave-Prozess versendet von dem der Root-Prozess den gerade abgespeicherten Member erhalten hat. Der Root-Prozess begibt sich dann wieder in die Ausgangssituation und wartet bis er von einem beliebigen Slave-Prozess den nächsten bewerteten Member empfängt. Dieser Zyklus wird dann so lange weiter geführt bis ein Konvergenzkriterium erfüllt ist oder der Nutzer *AutoOpti* abbricht.

Die hier beschriebene grundsätzliche Wirkweise von *AutoOpti* ähnelt somit den herkömmlichen Multi Objective Optimierungsprogrammen. Im Folgenden werden einige Merkmale beschrieben, die *AutoOpti* von üblichen Optimierungs-tools unterscheidet.

Bei der Umformulierung der Zielfunktionen ist darauf zu achten, dass sich diese nur aus denjenigen aerodynamischen/strukturmechanischen Parametern (flowparam) bilden lässt welche in der Datenbasis abgespeichert wurden. Aus diesem Grund, wurde in Kapitel 2.6 darauf hingewiesen möglichst alle physikalisch wichtigen Parameter abzuspeichern.

Behandlung von Nebenbedingungen:

AutoOpti kann für alle physikalischen Größen (flowparam) und für alle Zielfunktionen (fitness) Gültigkeitsintervalle berücksichtigen in denen die errechneten Größen liegen sollen. Falls ein bewerteter Member nicht vollständig dem so definierten Gültigkeitsbereich liegt, wird der Abstand zum Gültigkeitsbereich (Bez.; dist2RegionOfInterest) als zusätzliches Gütekriterium herangezogen. Der Begriff der Dominanz (Definition 3.1) wird in diesem Fall wie folgt erweitert:

Definition 3.1 modifiziert:

f_1, \dots, f_k seien die k Zielfunktionen einer multi-objective Optimierung mit Nebenbedingungen. Man sagt, dass ein bewerteter Member M_1 einen Member M_2 **dominiert**, falls dies bzgl. aller Zielfunktionen gilt (*bei mindestens gleiche Gültigkeit*) oder die Entfernung zum Gültigkeitsbereich von M_1 kleiner als von M_2 ist:

$$M_1 \prec M_2 \Leftrightarrow \left(\begin{aligned} & f_i(M_1) \leq f_i(M_2) \forall i \in \{1, \dots, k\} \wedge \exists j \in \{1, \dots, k\} : f_j(M_1) < f_j(M_2) \wedge \\ & \text{dist2RegionOfInterest}(M_1) \leq \text{dist2RegionOfInterest}(M_2) \end{aligned} \right) \\ \vee \text{dist2RegionOfInterest}(M_1) < \text{dist2RegionOfInterest}(M_2)$$

Dies bedeutet, dass die Erfüllung aller Nebenbedingungen das wichtigste Kriterium für die Bestimmung des Paretoranges ist und somit ausschlaggebend dafür ist ob ein Member als Elternteil selektiert wird.

Nebeneingang:

Um die Evolutionsstrategie zu beschleunigen (siehe Kapitel 3.2) und um dem Designer die Möglichkeit zu geben eine laufende Optimierung zu beeinflussen, wurde in *AutoOpti* der sog. Nebeneingang implementiert (siehe Bild 3.2). In diesen Nebeneingang kann der Nutzer (oder andere externe Programme) zur Laufzeit der Optimierung Datensätze freier Optimierungsparameter hinterlegen. Der Root-Prozess überprüft den Nebeneingang und überschreibt ggf. den mittels Vererbungsoperatoren erzeugten Member bevor dieser verschickt wird. Man kann also die folgende „Vorfahrtsregel“ aufstellen: Nebeneingang vor Vererbung.

Diese Strategie, *AutoOpti* während der Laufzeit steuern zu können hat sich bereits in vielen Optimierungen als äußerst erfolgreich erwiesen

Initialisierung, Versuchsplanung (DOE) und variable Parametertiefe

Die Initialisierung der Optimierung, d.h. die Belegung der ersten N Member kann auf unterschiedliche Arten erfolgen:

- zufällig
- durch Mutation eines initialen Members
- mittels eines Versuchsplans (DOE = **D**esign **O**f **E**xperiments)
- durch eine Vorooptimierung ohne CFD/FEM zur Sicherstellung der geometrischen Restriktionen

Ziel der Initialisierung sollte es sein ein möglichst breites Spektrum sinnvoller Member bereit zu stellen um somit einen größtmöglichen Parameterbereich bereits zu Beginn der Optimierung abzudecken. Üblicherweise werden für solche Zwecke sog. Versuchspläne durchgeführt. In *AutoOpti* ist ein Latin Hyper Cube Verfahren implementiert welches auch in hochdimensionalen Räumen anwendbar ist und zu den raumfüllenden Versuchsplänen gehört (siehe [Becker/Lawerenz 2008] Kapitel 5.1.3). Dieses Verfahren erscheint jedoch nur dann sinnvoll wenn zumindest ein gewisser Prozentsatz der so erzeugten Parametersätze die Prozesskette

erfolgreich durchläuft. Dies muss für die hier behandelten aerodynamischen Optimierungen jedoch nicht der Fall sein. Je nach Wahl der freien Parameter und der aerodynamischen Randbedingungen in den einzelnen Betriebspunkten kann es leicht vorkommen, dass von 100 mittels Latin Hyper Cube erzeugten Members keiner die geometrischen Restriktionen übersteht. Daher durchläuft *AutoOpti* in diesen Fällen eine Voroptimierung, in der die unterschiedlichen Parametersätze nur danach bewertet werden ob die geometrischen Restriktionen erfüllt sind oder nicht. Aus der Menge der so erzeugten Geometrien wird dann eine möglichst raumfüllende Teilmenge zur Initialisierung gewählt.

Auch dieses Verfahren scheitert jedoch wenn die Konvergenz des CFD-Verfahrens nicht für viele dieser unterschiedlichen Geometrien gewährleistet ist.

Des Weiteren kann die Parametertiefe mit dem gleichen Ziel (anfänglich breit gestreute Suche im Parameterraum) dynamisch gestaltet werden. Jeder freie Parameter eines Members besitzt die folgenden Attribute:

- den Parameterwert als reelle Zahl
- den Parameterwert als diskreten Wert
- die zugehörige Varianz, die der zugehörigen Mutationsschrittweite entspricht
- Ober- und Untergrenzen zum Festlegen des Intervallbereichs.

Neben der reellwertigen Darstellung gehört folglich zu jedem Parameter eine natürliche Zahl. Das Parameterintervall ist in diskrete Werte unterteilt und die Parametertiefe, also die Anzahl der zulässigen diskreten Werte, wird über eine Variable gesteuert. So bedeutet z. B. eine Parametertiefe von 4, dass das Parameterintervall in 2^4+1 Stützstellen unterteilt wird. Im Vererbungsschritt sind nur diese diskreten Stützstellen gültige Werte für den jeweiligen Parameter. Die zunächst niedrige Parametertiefe wird im Verlaufe der Optimierung sukzessive erhöht. Durch die anfänglich grobe Rasterung soll eine möglichst gute und gleichmäßige Abdeckung des Parameterraums erreicht werden. Mit jeder Veränderung der Parametertiefe müssen die zugehörigen ganzen Zahlen der Parameter aller Individuen angepasst werden.

3.2 Metamodelle

Ersatzmodelle/Metamodelle sind Funktionen, welche zur Approximation bzw. Interpolation von Daten (Punkte im Parameter-/Fitnessraum) verwendet werden. Dazu werden die Funktionsparameter so eingestellt, dass sie die Daten möglichst optimal wiedergeben. Im Idealfall wird eine wirklichkeitstreue Darstellung der Zielfunktion erreicht, so dass den Mitgliedern realitätsnahe Fitnesswerte zugeordnet werden können. Es können folglich gezielt Mitglieder mit niedrigen Fitnesswerten erzeugt werden.

Ersatzmodelle/Metamodelle werden mittels der bereits bewerteten Member erzeugt und dienen dem Zweck die Zielfunktionswerte an bisher unbekannten Positionen im Parameterraum zu schätzen. Da sie somit Approximationen von Simulationsergebnissen sind, die Ihrerseits ja auch nur ein Modell der Wirklichkeit darstellen, werden sie als Metamodelle bezeichnet (weitere übliche Bezeichnungen sind: Antwortfläche, Ersatzmodell, response surface, Approximationsmodelle, surrogate models, etc).

Die grundlegende Idee der Beschleunigung der ursprünglichen Optimierung durch den Einsatz von Ersatzmodellen ist leicht zu erklären: Die Prozesskette, wie Sie in Kapitel 2 beschrieben wurde, stellt aus Sicht des Optimierers lediglich eine Abbildung im folgenden Sinne dar:

$$\Phi: \mathbb{R}^N \rightarrow \mathbb{R}^K, \text{ mit } N = \#\{\text{freie Variablen}\} \text{ und } K = \#\{\text{Zielfunktionswerte}\}$$

Ein trainiertes Ersatzmodell ist demnach nichts anderes als eine Approximation $\phi: \mathbb{R}^N \rightarrow \mathbb{R}^K$ dieser Abbildung (siehe z.B. Bild 3.4 für eine Krigingapproximation). Wenn nun wie im vorliegenden Fall von CFD und FEM Simulationen eine Funktionsauswertung der originalen Abbildung Φ sehr zeitaufwendig ist, und eine Auswertung der Approximation ϕ vergleichsweise schnell ist, ist das Beschleunigungs-Potential für die Optimierung offensichtlich.

Das Zusammenspiel von Metamodellen und der gleichzeitig laufenden Optimierung ist in Bild 3.3 dargestellt. Die Metamodelle bilden ein eigenes parallelisiertes Programmpaket welches über die Module Datenbasis und Nebeneingang, welche in Kapitel 3.1 erläutert wurden, mit der originalen Optimierung kommuniziert. Die Hauptaufgaben dieses eigenständigen Programms liegen im Training geeigneter Modelle und der darauf folgenden Optimierung unter Verwendung der trainierten Modelle. Aus dieser temporären Optimierung auf den Ersatzmodellen werden dann besonders viel versprechende Member ausgewählt und der originalen Optimierung über den Nebeneingang zur *exakten* Zielfunktionsauswertung und darauf folgender Abspeicherung in die Datenbasis übergeben.

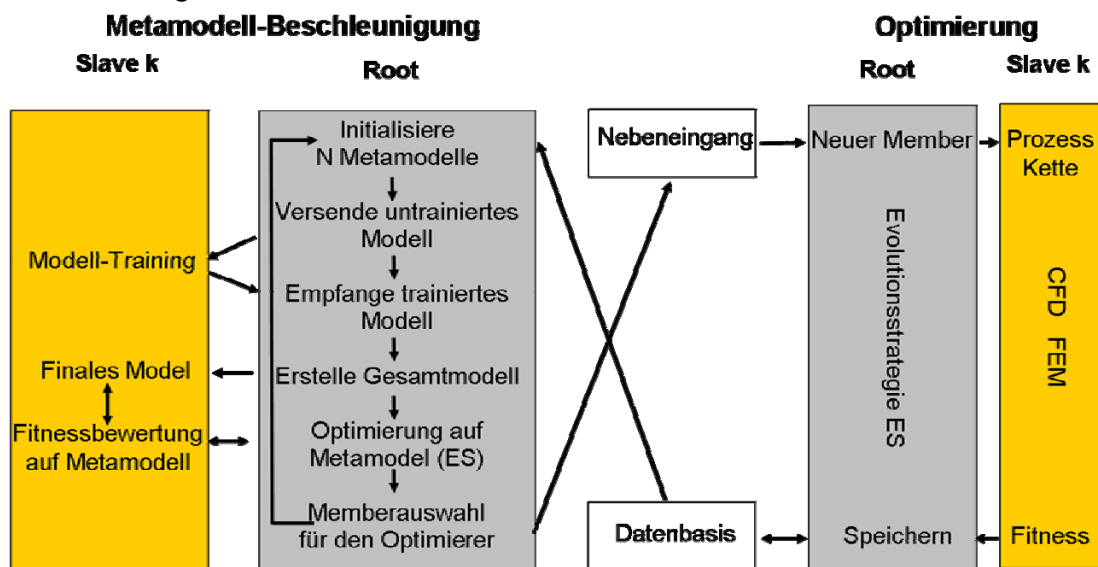


Bild 3.3: Flussdiagramm: Zusammenspiel AutoOpti-Metamodell

Bevor in den folgenden Kapiteln näher auf das Training der in *AutoOpti* implementierten Antwortflächen eingegangen wird, seien hier noch einmal die allgemeinen Gütekriterien für Ersatzmodelle erläutert.

Generalisierung: Ein Kriterium für die Güte von Ersatzmodellen

Generalisierung bedeutet das vorhandene Wissen so weit zu verallgemeinern, dass es auf neue Sachverhalte angewendet werden kann.

Beispielsweise müssen zur Schrifterkennung die Charakteristika der einzelnen Buchstaben erkannt werden, damit die individuelle Handschrift jedes Menschen gelesen werden kann. Andernfalls müsste der persönliche Schreibstil jedes Menschen gemerkt werden, um dessen Handschrift zu lesen. Neben dem damit verbundenen Aufwand, läge der große Nachteil darin, dass unbekannte Handschriften gar nicht gelesen werden könnten.

Das Beispiel der Schrifterkennung zeigt, dass für eine gute Generalisierung nicht zwangsläufig die vorhandenen Daten (hier die einzelnen Handschriften) gut wiedergegeben (erkannt) werden müssen. Übertragen auf die Ersatzmodelle bedeutet dies, dass approximierende Modelle teilweise eine bessere Generalisierung als Interpolierende besitzen. Dies ist besonders der Fall, wenn die Daten fehlerhaft sind. Beispielsweise, wenn sie aus einer Messung stammen, welche durch Störungen verfälscht wurde. Führt eine nahezu exakte Darstellung der Daten zu einer Verschlechterung der Generalisierung, wird von Over-fitting gesprochen. Innerhalb der automatischen Optimierung beschreibt Generalisierung die Fähigkeit, aus validierten Geometrien Zusammenhänge zu detektieren, um Aussagen über neue Geometrien treffen zu können.

Ein Weg zu einer guten Generalisierung

Generalisierung beschreibt was die Ersatzmodelle leisten sollen. Es wird aber keine Anleitung geboten, wie eine gute Generalisierung erreicht werden kann. Dies ist auch nicht möglich, da die Güte der Generalisierung nur bestimmt werden kann, wenn die reale Zielfunktion bekannt ist. Beispielsweise können Datenpunkte die auf einer Geraden liegen auch durch eine Sinuskurve, dargestellt werden, in dem die Frequenz der Sinusschwingung angepasst wird. Ist die erzeugende Funktion nun tatsächlich die Sinuskurve war die Wahl des Ersatzmodells richtig. Wurden die Punkte hingegen durch eine Gerade erzeugt, ist die lineare Approximation zu bevorzugen. Nun ist die erzeugende Funktion aber nicht bekannt. Somit gilt es abzuwägen, welche Erklärung plausibler ist. Dazu ist es notwendig den Begriff der *Plausibilität* genauer zu formulieren.

Plausibilität im Sinne von Ockhams Rasiermesser:

Die Genialität einer Konstruktion liegt in ihrer Einfachheit. Kompliziert bauen kann jeder. (Sergey P. Koroljew). Dieses Prinzip, auch bekannt als Ockhams Rasiermesser, bezeichnet das Sparsamkeitsprinzip in der Wissenschaft. Es besagt, dass von mehreren Theorien, die den gleichen Sachverhalt erklären, die einfachste zu bevorzugen ist.

Auf die Ersatzmodelle übertragen bedeutet dieses Sparsamkeitsprinzip, dass neben einer möglichst guten Wiedergabe der Daten (Erklärung des Sachverhalts), auch eine geringe Modellkomplexität erreicht werden sollte (Einfachheit der Theorie). Die Modellkomplexität kann entweder durch die Wahl des Ersatzmodells oder in einer Zielfunktion während des Trainings der Ersatzmodelle berücksichtigt werden (siehe Kapitel 3.2.2).

In den folgenden Kapiteln werden nun zwei unterschiedliche Metamodelle erläutert, die zum Teil innerhalb dieses Projektes entwickelt wurden. Zusätzlich steht noch ein Modul *Polynomiale Antwortflächen* zur Verfügung, welches an der Universität Kassel entwickelt wurde und in [Becker/Lawerenz 2008] beschrieben ist. Polynomiale Antwortflächen haben jedoch den Nachteil in hochdimensionalen Parameterräumen zu viele Stützpunkte zu benötigen (siehe Kapitel 5.2 in [Becker/Lawerenz 2008]).

In den folgenden Kapiteln 3.2.1 und 3.2.2 soll zuerst die Modellbildung und das Training der Modelle beschrieben werden. Im darauf folgenden Kapitel 3.3.3 werden die Modellmittelung und die Optimierung auf den Ersatzmodellen besprochen.

3.2.1 Kriging

Unter Kriging versteht man ein aus der Geostatistik stammendes Verfahren, mit dem man Werte an Orten, für die keine Probe vorliegt durch umliegende Messwerte interpolieren oder auch approximieren kann. Das Verfahren wurde von dem südafrikanischen Geostatistiker Daniel Krige (1952) entwickelt und später nach ihm benannt. Der wesentliche Vorteil gegenüber einfacheren Methoden wie beispielsweise der inversen Distanzgewichtung ist die Berücksichtigung der räumlichen Korrelationen der Messwerte.

Das in *AutoOpti* benutzte Kriging-Verfahren lässt sich wie folgt beschreiben:

Sei M die Anzahl ausgewerteter Member in der Datenbasis und N die Anzahl freier Variablen. Ein Vektor $\vec{x} = (x_1, \dots, x_N)$ bezeichnet einen Punkt im Parameterraum, die Vektoren $\vec{x}_i = (x_{i,1}, \dots, x_{i,N})$ $i=1, \dots, M$ bezeichnen die freien Parameter der bewerteten Member, der Vektor $\vec{y}_s = (y_1, \dots, y_M)$ bezeichnet für eine Zielfunktion den Vektor der Zielfunktionseinträge aller bisher bewerteter Member.

Die Korrelationsfunktion zwischen zwei Membern wird mit Cor , bezeichnet und ist in Gleichung (3) definiert. Es handelt sich dabei um eine Schar von Exponentialfunktionen mit $2 \cdot N$ Formvariabel sog. Hyperparametern Θ_k und p_k , die während des Trainings auf die Daten angepasst werden müssen.

Gleichung (4) zeigt die Korrelationsmatrix der bewerteten Member die in diesem Zusammenhang auch als Messwerte bezeichnet werden. In diese Matrix geht ein weiterer Hyperparameter λ als Diagonalaufschlag ein. Dieser Parameter dient als Regularisierungsparameter für die Korrelationsmatrix und beeinflusst das Interpolationsverhalten des Modells: Ist $\lambda=0$ werden die Messwerte interpoliert sonst approximiert.

$$\hat{y}(\vec{x}) = \hat{\beta} + \vec{r}^T(\vec{x}) \cdot \vec{R}^{-1}(\vec{\Theta}, \vec{p}, \lambda) \cdot (\vec{y}_s - \hat{\beta} \cdot \vec{1}) \quad , \vec{y}_s = (y_1, \dots, y_M) \quad (1)$$

$$\hat{\beta} = \frac{\vec{1} \cdot \vec{R}^{-1}(\vec{\Theta}, \vec{p}, \lambda) \cdot \vec{y}_s}{\vec{1} \cdot \vec{R}^{-1}(\vec{\Theta}, \vec{p}, \lambda) \cdot \vec{1}} \quad (2)$$

$$Cor(\vec{x}_i, \vec{x}_j) = \exp\left(-\sum_{k=1}^N \Theta_k \left|x_{i,k} - x_{j,k}\right|^{p_k}\right) \quad (3)$$

$$R_{i,j}(\vec{\Theta}, \vec{p}, \lambda) = Cor(\vec{x}_i, \vec{x}_j) + \lambda \cdot \delta_{i,j} \quad (4)$$

$$\vec{r}^T(\vec{x}) = (Cor(\vec{x}, \vec{x}_1), \dots, Cor(\vec{x}, \vec{x}_N))^T \quad (5)$$

$$\sigma^2 = M \cdot (\vec{y}_s - \hat{\beta} \cdot \vec{1}) \cdot \vec{R}^{-1}(\vec{\Theta}, \vec{p}, \lambda) \cdot (\vec{y}_s - \hat{\beta} \cdot \vec{1}) \quad (6)$$

$$MinLik = \sigma^2 \cdot \det(\vec{R}(\vec{\Theta}, \vec{p}, \lambda))^{1/M} \quad (7)$$

$$s^2(\vec{x}) = \sigma^2 \cdot \left[1 + \lambda - \vec{r}^T \cdot \vec{R}^{-1}(\vec{\Theta}, \vec{p}) \cdot \vec{r} + \frac{(1 - \vec{1} \cdot \vec{R}^{-1}(\vec{\Theta}, \vec{p}, \lambda) \cdot \vec{1})^2}{\vec{1} \cdot \vec{R}^{-1}(\vec{\Theta}, \vec{p}, \lambda) \cdot \vec{1}} \right] \quad (8)$$

Der beste Schätzer für den Erwartungswert der Zielfunktionen wird mit $\hat{\beta}$ (2) bezeichnet, der Vektor $\vec{r}(\vec{x})$ (5) ist der Korrelationsvektor zwischen einem neuen Punkt \vec{x} und den Messwerten $\vec{x}_i = (x_{i,1}, \dots, x_{i,N})$. Die globale Modellvarianz σ^2 berechnet sich aus Gleichung (6)

und die Unsicherheit bzw. Varianz s^2 der Vorhersage oder Schätzung $\hat{y}(\vec{x})$ (1) berechnet sich aus Gleichung (8).

Gleichung (1) sollte man wie folgt interpretieren: Die Vorhersage des Funktionswertes an einem Punkt ergibt sich aus der Summe eines Erwartungswertes $\hat{\beta}$ und einer lokal gewichteten Linearkombination der Abweichungen $(\vec{y}_s - \hat{\beta} \cdot \vec{1})$ aller Messwerte von diesem Erwartungswert.

Der Gewichtungsvektor der Linearkombination $\vec{g}(\vec{x}) = \vec{r}^T(\vec{x}) \cdot \vec{R}^{-1}(\vec{\Theta}, \vec{p}, \lambda)$ aus Gleichung (1), der angibt wie stark ein Messwert die Schätzung beeinflusst, sichert dabei den Kompromiss aus zwei Anforderungen:

Der i-te Messwert wird über den Faktor $\vec{r}(\vec{x})$ umso stärker gewichtet je stärker er mit dem auszuwertenden Punkt \vec{x} korreliert ist.

Der zweite Faktor $\vec{R}^{-1}(\vec{\Theta}, \vec{p}, \lambda)$ sorgt dafür, dass räumliche Korrelationen zwischen den Messwerten berücksichtigt werden.

Die Herleitung der Gleichungen (2), (6), (7) sowie die Schätzung der Parameter Θ_k und p_k ergibt sich aus den folgenden Betrachtungen:

Man versucht die Wahrscheinlichkeit zu maximieren, dass $\vec{y}_s = (y_1, \dots, y_M)$ einer multivariaten Normalverteilung mit dem skalaren Erwartungswert $\hat{\beta}$ und der Kovarianzmatrix $\vec{R}^{-1}(\vec{\Theta}, \vec{p}, \lambda)$ (wobei die Kovarianzen gemäß Gleichung (3) über räumliche Abstandsfunktionen gebildet werden) als Stichprobe entnommen wurde. Die Dichtefunktion dieser Normalverteilung ist gegeben durch:

$$p(\vec{y}, \vec{\Theta}, \vec{p}, \lambda) = \frac{1}{(2\pi)^{N/2} \det(\vec{R}^{-1}(\vec{\Theta}, \vec{p}, \lambda))^{1/2}} \exp\left((\vec{y} - \hat{\beta} \cdot \vec{1}) \cdot \vec{R}^{-1}(\vec{\Theta}, \vec{p}, \lambda) \cdot (\vec{y} - \hat{\beta} \cdot \vec{1})\right)$$

Nach dem Maximum-Likelihood Kriterium muss die rechte Seite obiger Gleichung für $\vec{y}_s = (y_1, \dots, y_M)$ maximiert werden. Logarithmiert man diese Gleichung und vernachlässigt additive Konstanten ist dies gleichbedeutend mit der Maximierung von:

$$l(\vec{\Theta}, \vec{p}, \lambda, \hat{\beta}) = -\frac{1}{2} \left[\log\left(\det(\vec{R}^{-1}(\vec{\Theta}, \vec{p}, \lambda))\right) + \left((\vec{y}_s - \hat{\beta} \cdot \vec{1}) \cdot \vec{R}^{-1}(\vec{\Theta}, \vec{p}, \lambda) \cdot (\vec{y}_s - \hat{\beta} \cdot \vec{1})\right) \right] \quad (9)$$

Der beste Schätzer für den Erwartungswert ergibt sich aus der partiellen Ableitung von Gleichung (9) $\frac{\partial l}{\partial \hat{\beta}} = 0$. Hieraus folgt Gleichung (2).

Des Weiteren lässt sich zeigen, dass die Maximierung von Gleichung (9) gleich bedeutend mit der Minimierung von Gleichung (7) ist. Die Schwierigkeit der Modellbildung liegt also in der der Minimierung von Gleichung (7). Dies ist ein Minimierungsproblem mit $2 \cdot N + 1$ Unbekannten. Die Dimension des Suchraumes hat sich gegenüber der ursprünglichen Optimierung also deutlich vergrößert. Zur Reduktion der freien Parameter werden alle Exponenten p_k in Gleichung (3) auf den gleichen Wert p gesetzt wodurch sich die Anzahl freier Parameter des Minimierungsproblems (7) auf $N + 2$ reduziert. Die Minimierung von (7) erfolgt dann innerhalb *AutoOpti* über ein Gradientenverfahren. Hierfür müssen die partiellen Ableitungen von Gleichung (7) nach $\vec{\Theta}, p, \lambda$ gebildet werden. Die Schrittweiten ergeben sich dann aus einem resilient backpropagation Verfahren wie man es auch zum Training neuronaler Netze verwenden kann.

Eine Schwierigkeit bei der Minimierung von Gleichung (7) liegt in der Determinantenbestimmung bzw. Aufstellung und Invertierung der Matrix $\vec{R}^{-1}(\vec{\Theta}, \vec{p}, \lambda)$ (benötigte numerische Operationen $\sim O(M^3)$) die bei kleiner werdenden Θ_k bzw. wachsendem Exponenten p schlecht konditioniert bzw. numerisch singular werden kann.

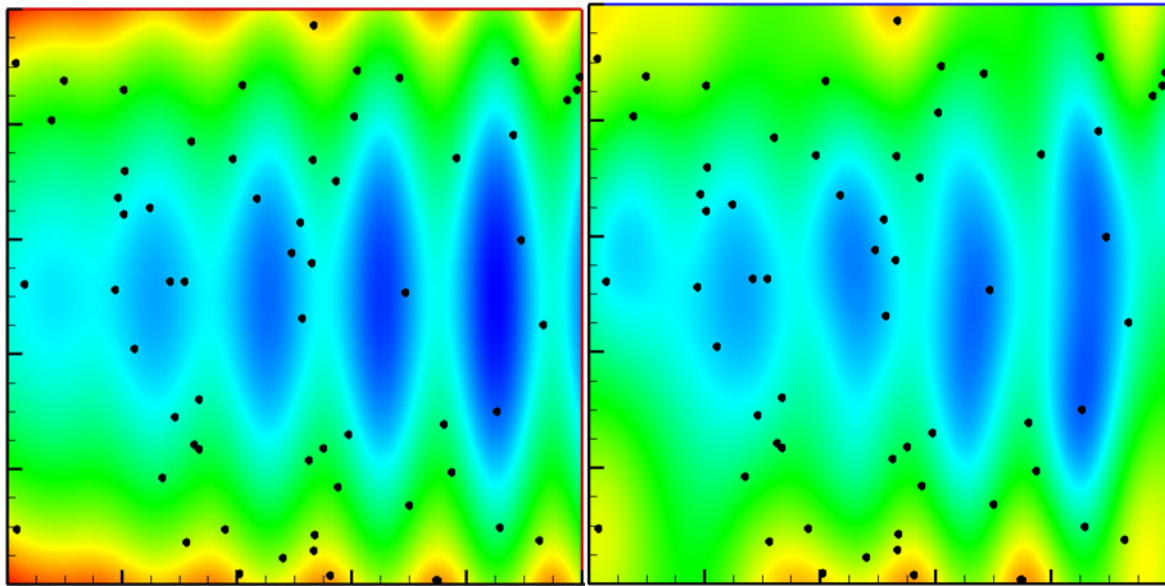


Bild 3.4 Links: Contour-plot einer reellwertigen Funktion Φ ; Rechts: Contour-plot der Krigingapproximation φ ; Die Punkte bezeichnen die Messwerte mit denen das Kriging-Modell trainiert wurde.

3.2.2 Neuronale Netze

Neuronale Netze bestehen aus einer Vielzahl einfacher Bausteine, genannt Neuronen. Diese Neuronen sind in verschiedenen Schichten gruppiert und miteinander verknüpft. Die Anzahl der Schichten, die Anzahl der Neuronen pro Schicht und die Art der Verknüpfung bestimmen die Topologie des Neuronalen Netzes.

In *AutoOpti* sind so genannte Feedforward-Netze implementiert. Eine Einführung in den Aufbau von Feedforward-Netzen und das Training mittels des Backpropagation-Algorithmus ist im Abschlussbericht des Vorhabens 1.1.2 (Förderkennzeichen 0327720A) skizziert.

Die dort beschriebene Basistechnologie wurde am Institut für Antriebstechnik weiter entwickelt.

Ein mögliches Ziel für das Training von Ersatzmodellen ist es, das Integral der Fehlerquadrate F zwischen Modellausgabe $y(x,w)$ und realer Funktion $f(x)$ zu minimieren (x bezeichnet die Punkte aus dem Designraum, w bezeichnet die Modellparameter und ist somit aus dem Gewichtsraum des Neuronalen Netzes).

$$F = \frac{1}{2} \int (f(x) - y(x, w))^2 dx$$

Da für die reale Funktion nur einzelne Auswertungen bekannt sind, wird nicht über den Fehler integriert, sondern die Fehlersumme über die Daten gebildet.

$$F_D = F_{\text{diskretisiert}} = \frac{1}{2} \sum_{i=1}^M (f(x_i) - y(x_i, w))^2$$

Das Prinzip von Ockhams Rasiermesser legt Nahe, dass die Modellkomplexität ein wichtiger Indiz für die Güte des Modells ist. Also ist es sinnvoll, dies im Training der neuronalen Netze zu berücksichtigen. Ein oft verwendetes Maß für die Netzkomplexität ist die Summe über die Quadrate der Gewichte (Weight-Decay):

$$F_w = \frac{1}{2} \sum_{i=1}^w w_i^2$$

Dabei ist zu beachten, dass die Trainingsdaten den Erwartungswert Null haben sollten, die Daten also beispielsweise standardisiert verwendet werden sollten. So sind auch Netze mit geringer Netzkomplexität nahezu erwartungstreu. Da nicht klar ist, wie die Netzkomplexität gegenüber der

Fehlerfunktion gewichtet werden soll, wird der Weight-Decay Term mit einem Regularisierungsterm λ multipliziert. Als neue Fehlerfunktion ergibt sich dann: $F = F_D + \lambda \cdot F_w$
 Natürlich stellt sich die Frage nach einer geeigneten Wahl von λ .

Bayesisches Lernen

Bayessches Lernen ist ein probabilistischer Ansatz zur Bestimmung des Regularisierungsterms. Anstatt eine Testmenge zu verwenden (Cross-Validation Ansatz), wird ein wahrscheinlichkeitstheoretischer Ansatz gewählt, welcher Rückschlüsse auf die Güte des Regularisierungsterms zulässt und zusätzlich eine Bewertung des Neuronalen Netzes erlaubt.

Wie der Name erkennen lässt, beruht Bayesisches Lernen auf der Regel von Bayes. Die Regel von Bayes gibt an, wie mit bedingten Wahrscheinlichkeiten gerechnet werden kann.

$$P(A|B) = \frac{P(A \cap B)}{P(B)} = \frac{\frac{P(A \cap B)}{P(A)} P(A)}{P(B)} = \frac{P(A) \cdot P(B|A)}{P(B)}$$

$P(A)$ wird als A-Priori Wahrscheinlichkeit, $P(A|B)$ als A-Posteriori Wahrscheinlichkeit bezeichnet. Im Folgenden werden Wahrscheinlichkeiten mit P und Dichtefunktionen von Wahrscheinlichkeitsverteilungen mit p bezeichnet

Regel von Bayes und Neuronale Netze

Die Regel von Bayes wird in Bezug auf die Neuronalen Netze so angewendet, dass die A-Priori-Wahrscheinlichkeit(-sdichte) für die Gewichte $p(w)$ in eine A-Posteriori-Wahrscheinlichkeit unter Berücksichtigung der Daten $p(w|D)$ überführt wird. Man möchte also die Wahrscheinlichkeit $p(w|D)$, dass die Daten die Gewichtseinstellung gut erklären, erhöhen.

$$p(w|D) = \frac{p(w) \cdot p(D|w)}{p(D)}$$

Da die Daten für ein Training gegeben sind, und sich somit $p(D)$ nicht beeinflussen lassen, muss man den Zähler in obiger Gleichung im Training maximieren. Dafür werden im Folgenden zwei Annahmen getroffen

1. Die Gewichte sind normalverteilt ($p(w)$)
2. Die Abweichung zwischen Daten und Ersatzmodell sind normalverteilt ($p(D|w)$).

Optimierung des Gewichtsvektors

Die Optimierung des Gewichtsvektors entspricht dem traditionellen Training mit weight-decay. Allerdings muss dieser Ansatz in den wahrscheinlichts-theoretischen Kontext des Bayesschen Lernen umformuliert werden.

Als Erstes gilt es, eine A-priori-Wahrscheinlichkeit $p(w)$ für den Gewichtsvektors zu finden. Hier wird eine Normalverteilung um Null mit Varianz $1/\alpha$ angenommen:

$$p(w_q) = \sqrt{\alpha} \cdot \varphi(\sqrt{\alpha} \cdot w_q); \quad \varphi(x) = \frac{1}{\sqrt{2\pi}} e^{-\frac{x^2}{2}}$$

φ ist die Dichtefunktion der Standardnormalverteilung. und w_q ist die q-te Komponente des Gewichtsvektors. α bestimmt die Größenordnung der Gewichte und wird deswegen auch als Hyperparameter bezeichnet. So bedeutet ein großes Alpha, dass die Gewichte nahe bei Null liegen. Die Verteilung $p(w)$ der Gewichte ergibt sich unter der Annahme, dass die Gewichte unabhängig sind, zu:

$$p(w) = \prod_{q=1}^W p(w_q) = \prod_{q=1}^W \sqrt{\frac{\alpha}{2\pi}} e^{-\frac{\alpha}{2} w_q^2} = \left(\frac{\alpha}{2\pi}\right)^{\frac{W}{2}} e^{-\alpha \sum w_q^2} = \left(\frac{\alpha}{2\pi}\right)^{\frac{W}{2}} \cdot e^{-\alpha \cdot F_w} = \frac{1}{C_\alpha} \cdot e^{-\alpha \cdot F_w} \quad (NN1)$$

Die Annahme, dass die Gewichte normalverteilt sind, führt dazu, dass auch die Trainingsdaten als normalverteilt angesehen werden. Dies gilt sowohl für den Parameter- als auch den Fitnessraum.

Außerdem ist die Unabhängigkeit der Gewichte in der Regel nicht zutreffend. Sind diese Voraussetzungen nicht gegeben, kann es zu Qualitätseinbußen kommen.

Für den zweite Term $\mathbf{p}(\mathbf{D}|\mathbf{w})$ aus der Bayes'schen Regel werden die Annahmen gemacht, dass die Fitnesswerte von einer stetigen Funktion $h(x)$ generiert werden und mit einem normalverteilten Rauschen ε (Fehler bei den gegebenen Fitnesswerten) behaftet sind. Die Fitnessfunktion t ergibt sich damit zu:

$$t(x) = h(x) + \varepsilon \Leftrightarrow \varepsilon = t(x) - h(x)$$

Das Rauschen bzw. die Abweichung vom Ersatzmodell sei durch eine Normalverteilung mit einer Varianz von $1/\beta$ gegeben:

$$p(\varepsilon) = \sqrt{\beta} \cdot \varphi(\sqrt{\beta} \cdot \varepsilon) = \frac{\sqrt{\beta}}{\sqrt{2\pi}} e^{-\frac{\beta \varepsilon^2}{2}}$$

Wird $h(x)$ mit der Ausgabe $y(x, w)$ eines Neuronalen Netzes identifiziert, ergibt sich der Dichtefunktionswert des Fehlers an einer Stelle x (entspricht einer Schätzung von $t|x$) zu:

$$p(t | x, w) = \sqrt{\beta} \cdot \varphi(\sqrt{\beta} (t(x) - y(x, w))) = \left(\frac{\beta}{2\pi} \right)^{\frac{1}{2}} \cdot e^{-\frac{\beta}{2} (y(x, w) - t(x))^2}$$

Für den Likelihood Term der ausgewerteten Daten ergibt sich:

$$p(D | w) = \prod_{i=1}^M p(t_i | x_i, w) = \left(\frac{\beta}{2\pi} \right)^{\frac{M}{2}} e^{-\frac{\beta}{2} \sum_{i=1}^M (y(x_i, w) - t_i)^2} = \frac{1}{C_\beta} e^{-\frac{\beta}{2} \sum_{i=1}^M (y(x_i, w) - t_i)^2} = \frac{1}{C_\beta} e^{-\beta F_D} \quad (\text{NN2})$$

A-Posteriori-Wahrscheinlichkeit der Gewichte

Die vorangehenden Annahmen haben nun die Voraussetzungen geschaffen, die A-Posteriori-Wahrscheinlichkeit $p(w|D)$ der Gewichte bei gegebenen Daten zu berechnen. Aus Gleichung (NN1) und (NN2) ergibt sich sofort:

$$p(w | D) = \frac{1}{p(D) \cdot C_\alpha \cdot C_\beta} \cdot e^{-\beta F_D - \alpha F_w} = \frac{\left(\frac{\beta}{2\pi} \right)^{\frac{M}{2}} \cdot \left(\frac{\alpha}{2\pi} \right)^{\frac{W}{2}}}{p(D)} \cdot e^{-(\beta F_D(w) + \alpha F_w(w))} =: g(\alpha, \beta, w) \quad (\text{NN3})$$

Die Normalisierungskonstanten sind unabhängig von den Gewichten. Um die A-Posteriori-Wahrscheinlichkeitsdichte $p(w|D)$ im Training (also durch Einstellung der Gewichte bei fest gewähltem α und β) zu maximieren, reicht es daher, den negativen Exponenten $F = \beta \cdot F_D + \alpha \cdot F_w$ zu minimieren. Dies ist gleichbedeutend mit dem oben skizzierten weight-

decay Ansatz, der Minimierung $F = F_D + \lambda \cdot F_w$ wobei $\lambda = \frac{\alpha}{\beta}$ gesetzt werden muss. Was bleibt

ist die *optimale* Kombination für α und β zu finden

Die Annahme, dass $t(x)-y(x, w)$ normalverteilt mit endlicher Standardabweichung ist, ist notwendig, um approximierende Neuronale Netze zu erzeugen. Andernfalls würde das Training einem Lernen ohne Weight-Decay entsprechen (sofern nicht die Nullfunktion gelernt werden muss, führt jedes mit Weight-Decay durchgeführte Training nur zu einer Approximation). Es ist aber auch zu sehen, dass der Likelihood Term größer wird, je näher die Netzausgaben $y(x, w)$ an den Zielwerten t liegen. Da β die Gesamtverteilung der Netzausgaben um die Fitnesswerte angibt, wird es, wie α als Hyperparameter bezeichnet.

Einstellung der Hyperparameter

Bei der Einstellung der Hyperparameter wird nun nicht mehr das Ziel verfolgt das Maximum der Dichtefunktion $p(w|D) = g(\alpha, \beta, w)$ (NN3) zu finden. Vielmehr wird durch eine geeignete Wahl von α und β versucht diejenige Zielfunktion aus der Funktionenschar zu erzeugen, die über alle Gewichte integriert die größte Wahrscheinlichkeit liefert:

$$\int p(w | D) dw = \frac{1}{p(D) \cdot C_\alpha \cdot C_\beta} \cdot \int e^{-\beta F_D - \alpha F_W} dw = \frac{\left(\frac{\beta}{2\pi}\right)^{\frac{M}{2}} \cdot \left(\frac{\alpha}{2\pi}\right)^{\frac{W}{2}}}{p(D)} \cdot \int e^{-(\beta F_D(w) + \alpha F_W(w))} dw \quad (\text{NN4})$$

Die Maximierung von (NN4) mittels α und β liefert dann in Kombination mit der Maximierung vom (NN3) mittels w , ein neuronales Netz (gekennzeichnet durch w_0 , α_0 und β_0), welches die durchschnittliche Plausibilität der Daten in einer Umgebung $U(w_0, \alpha_0, \beta_0)$ maximiert.

Im Gegensatz dazu, könnte die Maximierung von (NN3) mittels w , α und β zu einem Netz führen, welches die gegebenen Daten mit einer punktuell höheren Plausibilität erklärt, jedoch bereits durch leichte Änderung der Gewichte (bzw. der Trainingsdaten) schlechte Approximationen liefert. Die im Folgenden skizzierte Vorgehensweise berücksichtigt somit die Robustheit des erzeugten neuronalen Netzes.

Die Maximierung von (NN4) fällt schwer, da sich das Integral analytisch nicht lösen lässt. Daher wird die Fehlerfunktion F durch Ihre Taylorreihe genähert. Dabei ist darauf zu achten, dass der Gewichtsvektor w so eingestellt wurde, dass F dort ein Minimum und somit $F'(w)=0$ gilt:

$$F(w) = \beta F_D(w) + \alpha F_W(w) \doteq F(w_{\min}) + \left[(w - w_{\min})^T F'(w_{\min}) = 0 \right] + \frac{1}{2} (w - w_{\min})^T \cdot A \cdot (w - w_{\min})$$

Hierbei bezeichnet A die Hessematrix von F

$$A_{i,j}(w) = \frac{\partial^2}{\partial w_i \partial w_j} F(w) = \beta \frac{\partial^2}{\partial w_i \partial w_j} F_D(w) + \alpha I = \beta H + \alpha I \quad (\text{NN5})$$

Die Tatsache, dass bei der Taylorapproximation auf einen optimierten Gewichtsvektor zurückgegriffen wird, führt dazu, dass es sich beim Bayes'schen Lernen um ein hierarchisches Optimierungskonzept handelt. Auf der untersten Stufe wird der Gewichtsvektor optimiert. Erst im Anschluss können die Hyperparameter angepasst werden. Unter Zuhilfenahme der Taylorapproximation kann obiges Integral gelöst werden (zum Verständnis siehe Bestimmung der Normierungskonstanten einer multivariaten Normalverteilung).

$$\frac{\left(\frac{\beta}{2\pi}\right)^{\frac{M}{2}} \cdot \left(\frac{\alpha}{2\pi}\right)^{\frac{W}{2}}}{p(D)} \cdot \int e^{-(\beta F_D(w) + \alpha F_W(w))} dw = \frac{\left(\frac{\beta}{2\pi}\right)^{\frac{M}{2}} \cdot \left(\frac{\alpha}{2\pi}\right)^{\frac{W}{2}}}{p(D)} \cdot (2\pi)^{W/2} \cdot \det(A)^{-1/2} \cdot e^{S(w_{\min})}$$

Maximiert wird nun der Logarithmus obiger Gleichung:

$$\begin{aligned} \log\left(\int p(w | D) dw\right) &= \frac{M}{2} \log(\beta) + \frac{W}{2} \log(\alpha) - \frac{1}{2} \log(\det(A)) + S(w_{\min}) + C \\ &= \frac{M}{2} \log(\beta) + \frac{W}{2} \log(\alpha) - \frac{1}{2} \log(\det(A)) - \beta F_D + \alpha F_W + C =: h(\alpha, \beta, w_{\min}) \end{aligned} \quad (\text{NN6})$$

Hierbei ist C eine Konstante die nicht von den Hyperparametern oder Gewichten abhängt. Das Maximum von (NN6) berechnet sich aus:

$$\begin{aligned} \frac{\partial h}{\partial \alpha} &= \frac{W}{2\alpha} - \frac{\partial}{\partial \alpha} \left(\frac{1}{2} \log(\det(A)) \right) - F_W = 0 \\ \frac{\partial h}{\partial \beta} &= \frac{M}{2\beta} - \frac{\partial}{\partial \beta} \left(\frac{1}{2} \log(\det(A)) \right) - F_D = 0 \end{aligned} \quad (\text{NN7})$$

Für das Differenzieren einer Determinante gilt bei invertierbaren Matrizen:

$$\partial(\det(A)) = \det(A) \cdot \text{Spur}(A^{-1} \cdot \partial A) \Rightarrow \partial(\log(\det(A))) = \text{Spur}(A^{-1} \cdot \partial A) \text{ und somit folgt:}$$

$$\begin{aligned}\frac{\partial h}{\partial \alpha} = 0 &\Leftrightarrow W - \alpha \cdot \text{Spur}(A^{-1} \cdot I) - 2\alpha F_W = 0 \Leftrightarrow \alpha = \frac{W - \alpha \cdot \text{Spur}(A^{-1} \cdot I)}{2F_W} \\ \frac{\partial h}{\partial \beta} = 0 &\Leftrightarrow M - \beta \cdot \text{Spur}(A^{-1} \cdot \nabla^2 F_D) - 2\beta F_D = 0 \Leftrightarrow \beta = \frac{M - \beta \cdot \text{Spur}(A^{-1} \cdot \nabla^2 F_D)}{2F_D}\end{aligned}\quad (\text{NN8})$$

Man beachte hierbei, dass die Matrix A nach Gleichung (NN5) von (α, β) abhängt. Die Fixpunktgleichung (NN8) lässt sich durch Anwendung des Banachschen Fixpunktsatzes iterativ lösen:

$$\begin{aligned}\alpha_{k+1} &= \frac{W - \alpha_k \cdot \text{Spur}(A^{-1})}{2F_W} \\ \beta_{k+1} &= \frac{M - \beta_k \cdot \text{Spur}(A^{-1} \cdot \nabla^2 F_D)}{2F_D}\end{aligned}\quad (\text{NN9})$$

Dies ist äquivalent zur im Folgenden hergeleiteten update-Regel:

Seien λ_i die Eigenwerte von βH (siehe Gleichung (NN5)) so folgt mit (NN7)

$$\begin{aligned}0 &= \frac{W}{2\alpha} - \frac{\partial}{\partial \alpha} \left(\frac{1}{2} \log \prod_{i=1}^W (\lambda_i + \alpha) \right) - F_W = \frac{W}{2\alpha} - \frac{\partial}{\partial \alpha} \left(\frac{1}{2} \sum_{i=1}^W \log(\lambda_i + \alpha) \right) - F_W \\ 0 &= \frac{M}{2\beta} - \frac{\partial}{\partial \beta} \left(\frac{1}{2} \log \prod_{i=1}^W (\lambda_i + \alpha) \right) - F_D = \frac{M}{2\beta} - \frac{\partial}{\partial \beta} \left(\frac{1}{2} \sum_{i=1}^W \log(\lambda_i + \alpha) \right) - F_D\end{aligned}$$

Da alle λ_i die Konstante β als Faktor enthalten gilt $d\lambda_i/d\beta = \lambda_i / \beta$ und somit:

$$\begin{aligned}0 &= \frac{W}{2\alpha} - \left(\frac{1}{2} \sum_{i=1}^W \frac{1}{\lambda_i + \alpha} \right) - F_W \\ 0 &= \frac{M}{2\beta} - \frac{1}{2\beta} \left(\sum_{i=1}^W \frac{\lambda_i}{\lambda_i + \alpha} \right) - F_D\end{aligned}$$

Mit $\gamma := \left(\sum_{i=1}^W \frac{\lambda_i}{\lambda_i + \alpha} \right) = W - \left(\sum_{i=1}^W \frac{\alpha}{\lambda_i + \alpha} \right)$; da $\left(\sum_{i=1}^W \frac{\lambda_i}{\lambda_i + \alpha} \right) + \left(\sum_{i=1}^W \frac{\alpha}{\lambda_i + \alpha} \right) = W$ folgt sofort:

$$\begin{aligned}\alpha_{neu} &= \frac{\gamma}{2F_W} \\ \beta_{neu} &= \frac{M - \gamma}{2F_D}\end{aligned}\quad (\text{NN10})$$

Eine Veränderung von (α, β) führt dazu, dass der vorliegende Gewichtsvektor w_{\min} nicht mehr im Minimum der Fehlerfunktion liegt. Dies ist aber eine der Voraussetzungen (siehe Taylorapproximation) bei der Entwicklung der Updateregeln. Nach jedem Update von (α, β) gemäß (NN10) müssen also die Gewichte mittels eines Optimierungsverfahrens (Quasi Newton BFGS Algorithmus) und eines geeigneten Backpropagation Algorithmus zur Gradientenbestimmung neu berechnet werden.

Neben der Ausgabe eines neuronalen Netzes $y(x_0)$, welche eine Approximation der Zielfunktion an einer beliebigen Stelle x_0 im Designraum darstellt, kann zusätzlich eine Varianz $s^2(x_0)$ (ähnlich zu Gleichung 8 Kapitel 3.2.1) angegeben werden:

$$s^2(x_0) = \frac{1}{\beta} + \vec{g}(x_0)^T A(x_0)^{-1} \vec{g}(x_0); \text{ mit } g(x_0)_i = \frac{\partial y}{\partial w_i}(x_0)\quad (\text{NN11})$$

3.2.3 Ersatzmodell-Mittelung und Optimierung

Die Optimierung auf den jeweiligen Ersatzmodellen erfolgt gemäß dem in Bild 3.3 dargestellten Flussdiagramm. Das Training der Modelle wurde in den letzten Kapiteln beschrieben. In diesem Kapitel soll es nun um die Mittelung unterschiedlicher Modelle und die Optimierung auf dem Ersatzmodell gehen. Dabei wird nicht mehr nach dem Typ der Antwortfläche (Kriging und/oder neuronales Netz) unterschieden.

Mittelung

Eine deutliche Verbesserung der Vorhersagesicherheit eines Ersatzmodells kann sich durch eine geeignete Mittelung über mehrere Modelle ergeben. Wie im Flussdiagramm in Bild 3.3. angedeutet wird eine gewisse Anzahl (gesteuert über Input-Parameter) von Modellen zufällig initialisiert und danach von den Slave-Prozessen trainiert. Unter der Voraussetzung, dass die einzelnen Modelle statistisch unabhängig sind (gewährleistet durch zufällige Initialisierung der Modellparameter und/oder zufällige Auswahl von Trainingsmustern), ergibt sich für die Varianz des durch Mittelung geschätzten Funktionswerts eine Reduktion gegenüber der Varianz jedes einzelnen Modells. Verdeutlicht sei dies für den Fall des arithmetischen Mittels von N statistisch unabhängigen Modellen X_i gleicher Varianz S^2 . Es gilt:

$$Var(\bar{X}) = Var\left(\frac{1}{N} \sum_{i=1}^N X_i\right) = \frac{1}{N^2} \sum_{i=1}^N Var(X_i) = \frac{S^2}{N}$$

Innerhalb von *AutoOpti* wird an jedem auszuwertendem Punkt diejenige Linearkombination der einzelnen Modelle gewählt, welche die gemittelte Varianz minimiert. Dies ist gleichbedeutend mit dem folgenden Minimierungsproblem:

Finde an einer Stelle \bar{x} die lokalen Gewichtungsfaktoren a_i , so dass das gemittelte Modell

$$\tilde{X}(\bar{x}) = \sum_{i=1}^N a_i \cdot X_i(\bar{x}) \text{ mit } Var(X_i(\bar{x})) = s_i^2(\bar{x}) \quad (\text{Kapitel 3.2.1 Gleichung (8) bzw. Gleichung (NN11)})$$

eine minimale Varianz $Var(\tilde{X}) = \sum_{i=1}^N a_i^2 \cdot s_i^2(\bar{x})$ aufweist. Beachte dabei die Nebenbedingung: $\sum a_i = 1$

$$\text{sei } f(a_i, \lambda) := \sum_{i=1}^N a_i^2 \cdot s_i^2(\bar{x}) + \lambda \cdot (\sum a_i - 1)$$

$$grad(f) = 0 \Rightarrow a_i = \frac{\frac{1}{s_i^2(\bar{x})}}{\frac{1}{s_1^2(\bar{x})} + \dots + \frac{1}{s_N^2(\bar{x})}} \wedge \lambda = -\frac{2}{\frac{1}{s_1^2(\bar{x})} + \dots + \frac{1}{s_N^2(\bar{x})}}$$

$$\Rightarrow Var(\tilde{X}) = \frac{1}{\left(\frac{1}{s_1^2(\bar{x})} + \dots + \frac{1}{s_N^2(\bar{x})}\right)^2} \sum \frac{1}{s_i^2(\bar{x})} = \frac{1}{\frac{1}{s_1^2(\bar{x})} + \dots + \frac{1}{s_N^2(\bar{x})}} \cdot \frac{1}{\frac{1}{s_i^2(\bar{x})}} \leq \frac{1}{s_i^2(\bar{x})} \forall i \in \{1, \dots, N\}$$

Es zeigt sich also, dass durch die geeignete Mittelung die Unsicherheit der Modellausgaben deutlich reduziert werden kann.

Expected Improvement Kriterium

Bei der Optimierung auf dem gemittelten Modell können unterschiedliche Ziele verfolgt werden. Das vorrangige Ziel sollte dabei natürlich die Beschleunigung der Optimierung durch die Erzeugung viel versprechender Kandidaten sein. Ein weiteres Ziel kann die Verbesserung des Modells darstellen, indem gezielt Kandidaten in Regionen mit größerer Vorhersagevarianz also

größerer Unsicherheit erzeugt werden, und zur *exakten* Berechnung im Nebeneingang hinterlegt werden.

Das sog. expected imrovment Kriterium welches im Folgenden beschrieben wird, sucht daher diejenigen Kandidaten mit der besten erwarteten Verbesserung und stellt einen Kompromiss aus den beiden genannten Zielen dar. Dieses Kriterium ist auf alle Ersatzmodelle anwendbar die neben der Vorhersage $\hat{y}(x)$ auch die Standardabweichung $S(x)$ bzw. die Varianz $S^2(x)$ für das Vertrauensniveau dieser Vorhersage liefern.

In *AutoOpti* berechnet sich die Varianz des gemittelten Modells $\tilde{X}(\vec{x}) = \sum_{i=1}^N a_i \cdot X_i(\vec{x})$ aus den Varianzen der einzelnen Modellvorhersagen $s_i^2(x)$ und den Abweichungen der einzelnen Modelle vom gemittelten Modell. Es ergibt sich also:

$$S^2(X_i(x)) = s_i^2(x) + (X_i(x) - \tilde{X}(x))^2$$

$$\Rightarrow S^2(\tilde{X}(x)) = \sum_{i=1}^N S^2(X_i(x)) \cdot \left(\frac{\frac{1}{s_i^2(\vec{x})}}{\frac{1}{s_1^2(\vec{x})} + \dots + \frac{1}{s_N^2(\vec{x})}} \right)^2$$

$$\Rightarrow S(\tilde{X}(x)) = \frac{1}{\frac{1}{s_1^2(\vec{x})} + \dots + \frac{1}{s_N^2(\vec{x})}} \sqrt{\sum_{i=1}^N \frac{S^2(X_i(x))}{(s_i^2(\vec{x}))^2}}$$

Im Folgenden wird von einer single objective Optimierung ausgegangen. Für die Erweiterung des expected imrovment Kriterium auf multi-objective Optimierungen siehe [Keane 2006]

Sei $\hat{y}(x)$ die Ersatzmodellvorhersage der Zielfunktion f an einer Stelle x , und $S(x)$ die prognostizierte Standardabweichung an dieser Stelle (siehe oben) Das in der Optimierung bisher gefundene Minimum der Zelfunktion f sei mit $f_{\min}(x)$ bezeichnet.

Das expected improvement Kriterium an einer Stelle x_0 ergibt sich aus dem Erwartungswert $E[]$ der Verbesserung (siehe [Keane 2006])

$$E[I[x_0]] = E[\max\{f_{\min} - \hat{y}, 0\}]$$

$$= (f_{\min} - \hat{y}(x_0))\phi(u) + S(x_0) \cdot \varphi(u) \text{ mit } u = \frac{\hat{y}(x_0) - f_{\min}}{S(x_0)}$$

Hierbei bezeichnet ϕ die Dichtefunktion der kumulierten Standardnormalverteilung Φ .

In *AutoOpti* wird dieses Kriterium bei der Auswahl geeigneter Member herangezogen. Sowohl bei der Selektion der Eltern während der Optimierungsiteration, als auch bei der finalen Auswahl viel versprechender Kandidaten wird versucht die gemeinsame erwartete Verbesserung zur Paretofront zu maximieren. Die Berechnung des Paretorangs erfolgt jedoch nach wie vor mittels der vorhergesagten Zielfunktionen.

4. Zielfunktionsformulierung

Der wichtigste Bestandteil jeder Optimierung ist (neben der Wahl der freien Parameter und deren Grenzen) die Wahl geeigneter Zielfunktionen und Nebenbedingungen, anhand derer verschiedene Konfigurationen bzw. Member gegeneinander bewertet werden können. Die Zielfunktionen müssen innerhalb von *AutoOpti* so definiert sein, dass sie ein Minimierungsproblem beschreiben. Dies stellt jedoch keine Einschränkung dar, da jeder zu maximierende Zielwert f durch Negation ($-f$) in ein Minimierungsproblem überführt werden kann, ohne damit die Komplexität der Funktion zu beeinflussen.

Die technische Umsetzung der Zielfunktionsformulierung innerhalb *AutoOpti* wurde bereits in Kapitel 2.6 besprochen und sei im Folgenden kurz rekapituliert:

In der Struktur *flowparam* werden alle physikalischen Parameter ρ_k abgespeichert die für einen Member von Interesse sind (Grund: restart-optionen). Diese ergeben sich üblicherweise aus den CFD und FEM-Rechnungen. Die Struktur *fitness* berechnet sich aus der Struktur *flowparam* und beinhaltet die Zielfunktionen f_i . Für die Belegung dieser Größen sind zwei C-Funktionen reserviert die vor jeder Optimierung vom Nutzer angepasst werden müssen.

Bei der Bewertung einer Konfiguration, d.h. bei der Bestimmung des Paretoranges, wird zuerst überprüft ob alle Nebenbedingungen erfüllt sind. Diese Nebenbedingungen überprüfen einzelne physikalische Parameter ρ_k auf die Einhaltung gewisser Schranken: $\rho_k \in [\rho_{k,\min}, \rho_{k,\max}]$ und wirken unmittelbar auf den Paretorang des Members (siehe: Kapitel 3.1 Behandlung von Nebenbedingungen).

Eine weitere Möglichkeit Nebenbedingungen zu berücksichtigen ist über die Implementierung von Straftermen in den Zielfunktionen gegeben (siehe Kapitel 4 in [Becker/Lawerenz 2008]). Diese beeinflussen dann jedoch die Komplexität der Zielfunktionen, was gerade im Hinblick auf das Training der Ersatzmodelle unerwünscht sein kann.

Die Anforderungen und Ziele der Verdichterauslegung sind zu vielfältig um hier ein generell gültiges *Kochrezept* zur Zielfunktionsformulierung angeben zu können. Vielmehr sollen einige in diesem Vorhaben gewonnene Erfahrungen Erwähnung finden, die sich auch auf zukünftige Verdichterauslegungen oder andere Optimierungsanwendungen übertragen lassen

- Man sollte zu Beginn jeder Optimierung bedenken, dass die gewählten Zielfunktionen und Nebenbedingungen höchstwahrscheinlich nicht alle Aspekte einer erfolgreichen Verdichterauslegung abdecken können. Daher sollte man zusätzlich möglichst alle physikalisch relevanten Parameter ρ_k abspeichern und somit einen verlustfreien Neustart der Optimierung mit variierten Zielen ermöglichen.
- Die Wahl zu vieler Zielfunktionen (vor dem Hintergrund möglichst viele Aspekte abdecken zu wollen) reduziert den Evolutionsdruck der Optimierung. Ein hoher Prozentsatz der erzeugten Konfigurationen wird Paretorang 1 besitzen. Daher wird sich die Paretofront nur langsam entwickeln. Die Erfahrungen innerhalb dieses Projektes haben gezeigt, dass mehr als drei (paarweise unkorrelierte) Zielfunktionen vermieden werden sollten.
- Die gewählten Zielfunktionen sollten unkorreliert formuliert werden. Paarweise korrelierte Zielfunktionen sollten zu einer einzigen Zielfunktion zusammengefasst werden.
- Die aerodynamischen Ziele für die einzelnen Betriebspunkte setzen sich typischerweise aus folgenden Größen zusammen welche ein Resultat der Simulation und ggf. der Mittelung im post-processing darstellen:

Massenstrom : \dot{m} ; Totaldruckverhältnis : Π_{tot} ; statisches Druckverhältnis : Π_{stat} ;
 Diffusionszahl : DF ; Totaltemperatur : T_{tot} ; Wirkungsgrad : η_{is} ; Abtrömwinkel : α_{exit} ;
 Totaldruckverlust : ω ; deHaller – Koeffizient; Machzahl Ma ; Dichte ρ

Es ist jedoch in vielen Fällen sinnvoll zusätzliche Größen zu definieren. Ein Beispiel hierfür ist der sog. reduzierte Wirkungsgrad welcher den Einfluss des Totaldruckverhältnisses auf den Wirkungsgrad durch Implementierung einer oberen Schranke restringiert. Ein weiteres Anheben des Totaldruckverhältnisses über diese Schranke wird im reduzierten Wirkungsgrad nicht mehr belohnt.

$$\eta_{\text{red}} = \left\{ \begin{array}{l} \frac{\frac{\pi_t^{\frac{1-\kappa}{\kappa}} - 1}{T_{t,\text{exit}} - 1}}{\frac{\pi_t^{\frac{1-\kappa}{\kappa}} - 1}{T_{t,\text{entry}} - 1}}; \text{ falls } \pi_t < \pi_{t,\text{max}} \\ \frac{\pi_{t,\text{max}}^{\frac{1-\kappa}{\kappa}} - 1}{T_{t,\text{exit}} - 1} \text{ sonst} \end{array} \right.$$

- Ein typisches aerodynamisches Optimierungsziel in der Verdichterauslegung besteht in der Maximierung der Pumpgrenze. Diese Größe ist in einem automatischen Prozess jedoch nur schwer und mit großem numerischem Aufwand ermittelbar (dies ist u.a. Ziel eines zukünftigen Projektes COORETEC-turbo Tranche 3, Vorhaben 1.1.5; FKZ 0327717C). Als Hilfslösung kann man jedoch einen Pumpgrenznahen Betriebspunkt in die Optimierung aufnehmen und diesen in TRACE Massenstrom-geregelt ansteuern (ein Beispiel findet sich im nachfolgenden Kapitel 5: OP1). Das Totaldruckverhältnis an diesem Punkt ist dann ein Maß für die Steilheit der Drehzahllinie und somit ein Hilfsmaß für die Pumpgrenze. Es ist jedoch abzuwägen wie nah dieser Betriebspunkt an der geforderten Pumpgrenze (definiert über den Durchsatz) gewählt wird, um Konvergenzprobleme der Strömungssimulationen bei einem hohen Prozentsatz der Konfigurationen zu vermeiden.

5. Optimierung mittels Trace/Calculix: Gegenläufiger Fan

In diesem Kapitel wird eine aerodynamische Optimierung mit strukturmechanischen Nebenbedingungen vorgestellt die mittels des Programmpaketes *AutoOpti* durchgeführt wurde. Diese soll als typisches Beispiel der multidisziplinären Axialverdichteroptimierung dienen. Gegenstand der hier beschriebenen Optimierung ist ein gegenläufiger Fan eines Hochbypass-Triebwerks. Die Optimierung fand innerhalb des EU-Projektes VITAL statt. Partner in diesem Projekt sind neben dem DLR: Snecma, Ciam, Cenaero und Comoti.

Die mittels *AutoOpti* erzeugte Fankonfiguration wird als CRTF2 bezeichnet und wurde von den Partnern ausgewählt in Blisk-Bauweise konstruiert und im 1.Quartal 2009 bei CIAM getestet zu werden.

Freie Optimierungsparameter-Wahl der Parametrisierung

In dieser Optimierung waren sowohl der Verlauf von Nabe und Gehäuse als auch die Geometrien beider Rotoren zur Optimierung frei gegeben. Es wurde ein initialer gegenläufiger Fan mit dem Bladegenerator des Instituts für Antriebstechnik (siehe Kapitel 2.1) entworfen. Dabei wurde im Hinblick auf die automatische Optimierung eine möglichst robuste Parametrisierung der Konstruktionsprofile über die variable Druckseite gewählt (siehe Kapitel 2.1.1 Parametrisierung2). Ausgehend von dieser initialen Konfiguration wurden über die Optimierungsfreigaben

Variationen der Geometrie ermöglicht. Im Folgenden seien die freien Optimierungsparameter erläutert.

Naben- und Gehäuseverlauf

Bild 5.1 zeigt den initialen Strakverlauf (durchgezogene Linien; rot Nabe; grün Gehäuse) mitsamt den Freigaben für die Optimierung. Jeder der Konstruktionspunkte auf Nabe und Gehäuse kann dabei in den angedeuteten Intervallen variiert werden. Der finale Kurvenverlauf ergibt sich dann durch einen interpolierenden B-Spline 3. Grades durch diese Strakpunkte. Der in Bild 5.1 fest gehaltene Punkt auf der Nabe (der stromauf und stromab freie Nachbarn hat) entspricht dabei der Lage der Vorderkante von Rotor1. Die Fixierung dieses Punktes ist baulichen Maßnahmen geschuldet.

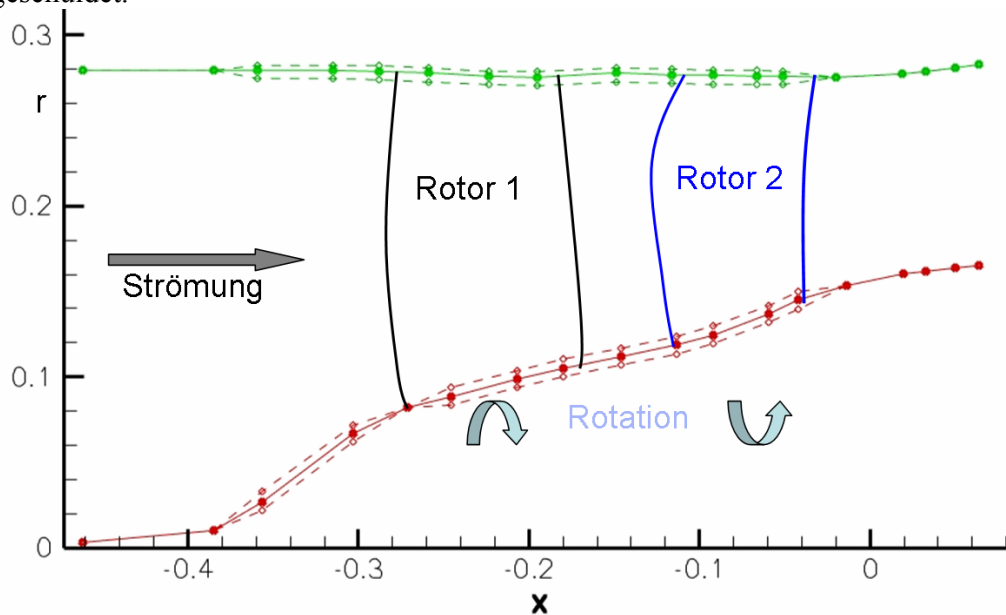


Bild 5.1: Strakverlauf mit Freigaben

Rotor1

Rotor1 wurde mit 7 Profilen konstruiert und besteht aus 9 Schaufelblättern. Für die Saugseitenkonstruktion wurden 5 Kontrollpunkte zugelassen. Die Druckseite wurde über die Vorgabe der strukturmechanisch relevanten Parameter aus Parametrisierung 2 (d_{max} , Füllungsgrad, etc) konstruiert. Die Fädelung erfolgte über die Vorgabe einer Vorder- und Hinterkantenlinie (Kapitel 2.1.2 Variante2) sowie Θ -shifts für die Konstruktionsprofile. Dabei wurden die Vorderkante und die Hinterkante mit jeweils 6 Punkten aufgelöst. Die für die Optimierung frei gegebenen Designparameter von Rotor1 sind in Bild 5.2 dargestellt.

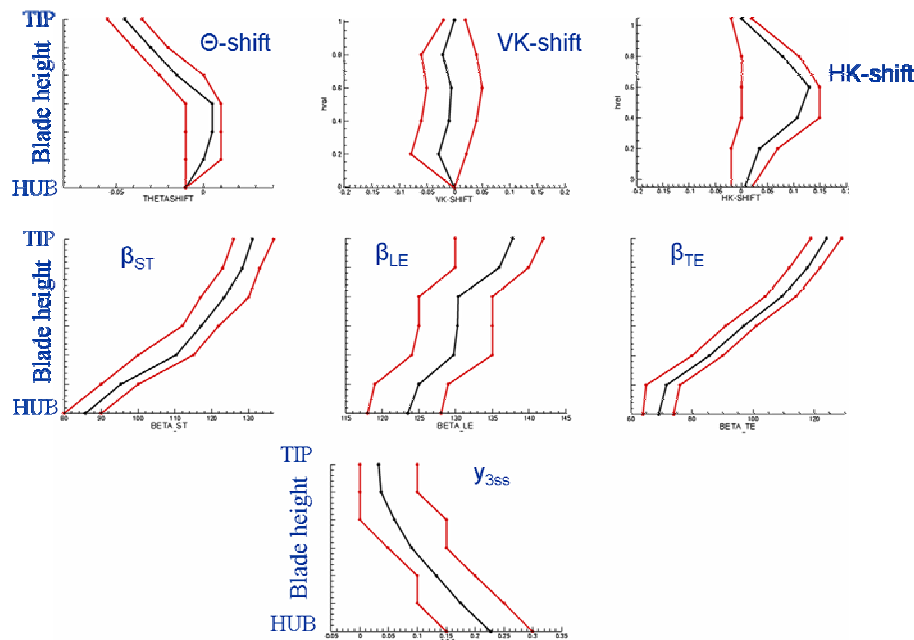


Bild 5.2: Rotor1 Freigaben: schwarz: initiales Design; rot Freigabe

Rotor2

Rotor2 wurde mit 7 Profilen konstruiert und besteht aus 11 Schaufelblättern. Für die Saugseitenkonstruktion wurden 7 Kontrollpunkte zugelassen. Die Druckseite wurde über die Vorgabe der strukturell relevanten Parameter aus Parametrisierung 2 (d_{max} , Füllungsgrad, etc) konstruiert. Die Fädelung erfolgte über die Vorgabe einer Vorder- und Hinterkantenlinie (Kapitel 2.1.2 Variante2) sowie Θ -shifts für die Konstruktionsprofile. Dabei wurden die Vorderkante und die Hinterkante mit jeweils 7 Punkten aufgelöst. Die für die Optimierung frei gegebenen Designparameter von Rotor2 sind in Bild 5.3 dargestellt.

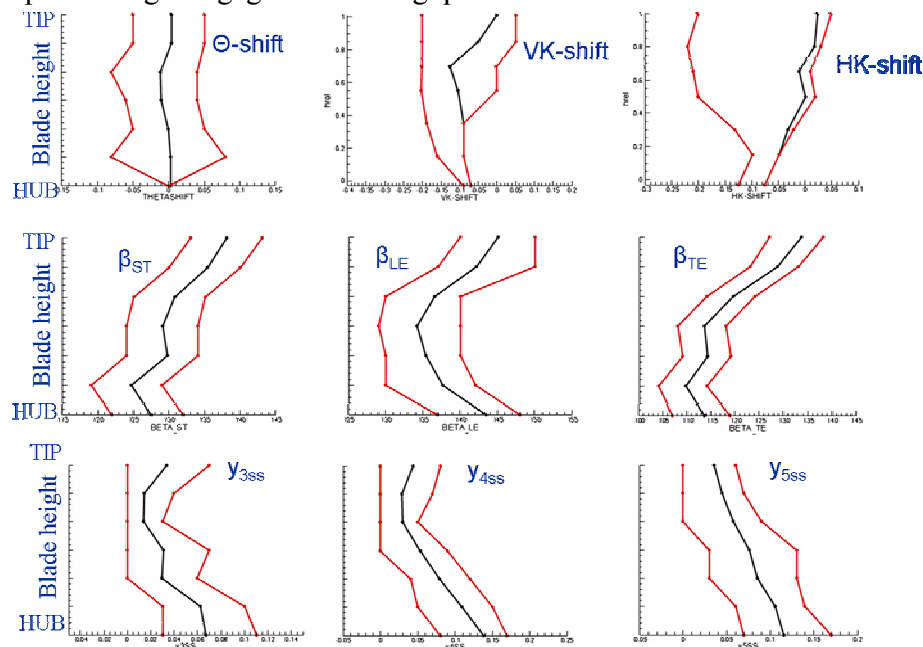


Bild 5.3: Rotor2 Freigaben

In Summe ergeben sich die in Tabelle 5.4 eingetragenen freien Parameter:

45 freie Designparameter für Rotor1; 62 freie Designparameter für Rotor2; 10 freie Designparameter für den Nabenverlauf und 11 freie Designparameter für den Gehäuseverlauf. Daraus ergibt sich insgesamt ein Design- oder Suchraum der Dimension 128.

Variable	Rotor1	Rotor2	Hub	Tip	Nr. of free variable
Theta-Shift	6	6			
β_{LE}	7	7			
β_{TE}	7	7			
β_{ST}	7	7			
L-Edge shape	5	7			
T-Edge shape	6	7			
y_{3ss}	7	7			
y_{4ss}	-	7			
y_{5ss}	-	7			
Σ	45	62	10	11	128

Tabelle 5.4: Gegenläufiger Fan CRTF2, Freigaben für die Optimierung

Betriebspunkte

Die in dieser Optimierung für die Aerodynamik berücksichtigten Betriebspunkte sind in Tabelle 5.5 dargestellt. Es handelt sich dabei um den Betriebspunkt OP0=ADP (Aerodynamische Design Punkt) auf der 100%-Drehzahllinie, einen Pumpgrenznahen Betriebspunkt auf der 100%-Drehzahllinie welcher mit OP1 bezeichnet wird, und den so genannten Approach Punkt OP2 auf der 50%-Drehzahllinie.

OP1 und OP2 werden dabei in TRACE über die Vorgabe des Massenstroms angesteuert, während in OP0 die klassische Randbedingung des statischen Gegendrucks gesetzt wird.

Betriebspunkt	Randbedingung TRACE	Zielgrößen	Restriktionen	Drehzahl
OP0=ADP	P_{stat} Statischer Gegendruck	η isentroper Wirkungsgrad	$m \in [m_{min}, m_{max}]$ $\pi_{tot} \in [\pi_{min}, \pi_{max}]$	100%
OP1	m Massenstrom		$\pi_{tot} \in [\pi_{min}, \infty]$	100%
OP2	m Massenstrom	η isentroper Wirkungsgrad		50%

Tabelle 5.5: Aerodynamisch relevante Betriebspunkte

Zielfunktionen-Restriktionen

Tabelle 5.5 zeigt die Ziele die für die einzelnen Betriebspunkte verfolgt werden:

OP0: Der isentrope Wirkungsgrad soll hier erhöht werden. Dies ist gleich bedeutend mit einer Effizienzsteigerung bzw. mit einer Treibstoffersparnis. Da dieser Betriebspunkt innerhalb von TRACE über den statischen Gegendruck eingestellt wird, ist darauf zu achten, dass sich der Punkt im Kennfeld des Verdichters nicht signifikant verschiebt. Daher wird darauf geachtet, dass Massenstrom und Totaldruckverhältnis in engen Grenzen bleiben (Siehe Kapitel 3.1 *Behandlung von Nebenbedingungen*).

OP1: Die Simulation dieses Betriebspunkts soll einen hinreichenden Arbeitsbereich des Verdichters garantieren. Dieser Betriebspunkt wird über die Vorgabe des Massenstroms gesteuert so dass bei Einhaltung einer minimalen Grenze für das Totaldruckverhältnis ein geforderter Pumpgrenzabstand garantiert werden kann.

OP2: Dieser Betriebspunkt ist repräsentativ für die Landeanflugphase und daher wichtig für die aeroakustische Bewertung des Triebwerks. Obwohl eine detaillierte aeroakustische Abschätzung des gegenläufigen Fans nur unter Berücksichtigung der instationären Strömung möglich ist, soll mit der Maximierung des Wirkungsgrades ein *aufgeräumtes* Strömungsfeld in diesem Betriebspunkt erreicht werden: Ein hoher isentroper Wirkungsgrad der stationären Lösung ist ein Indiz für ein vorteilhaftes (instationäres) aeroakustisches Verhalten.

Strukturmechanische Restriktionen:

Rotor1 und Rotor2 wurden strukturmechanisch mittels Calculix bewertet (statische Rechnung am ADP). Die maximal zulässige von Mises Spannungen wurde dabei sowohl für Rotor1 als auch für Rotor2 auf 460MPa gesetzt.

Resultate

Die Optimierung lief auf dem Linux-Cluster des Instituts für Antriebstechnik des DLR in Köln. Es wurden ca. 1050 unterschiedliche Member erzeugt von denen ca. 650 Konfigurationen in allen 3 Betriebspunkten konvergierten. Die Member wurden (nachdem 150 konvergente Member gerechnet waren) über Ersatzmodelloptimierungen erzeugt. Hier wurde ein gemitteltes Kriging genutzt. Die Optimierung wurde nach 650 konvergenten Membern (also 1950 TRACE-Rechnungen und 650 Calculix-Rechnungen) unterbrochen, da alle aerodynamischen und strukturmechanischen Ziele und Nebenbedingungen erreicht waren. Danach wurde ein Member (Bezeichnung Memb755) ausgewählt der hier kurz vorgestellt wird und im Frühjahr 2009 bei CIAM experimentell überprüft werden soll.

Bild 5.4 zeigt das Kennfeld (bei 100% Drehzahl) von Memb755 (in grün) und zwei weiteren Designs. Im linken Teil ist das Totaldruckverhältnis über dem Massenstrom; im rechten Teil das Totaldruckverhältnis über dem Wirkungsgrad aufgetragen.

Bei dem in orange dargestellten Design handelt es sich um eine Snecma-Konfiguration. Diese trifft jedoch nicht den gewünschten Betriebspunkt OP0 auf der Arbeitslinie. Des Weiteren zeigt dieses Design einen deutlich geringeren Wirkungsgrad als Memb 755.

Die blaue Kennlinie stammt von einer bereits experimentell überprüften Konfiguration, welche jedoch eine höhere Schaufelzahl aufweist (Rotor1 10Schaufeln, Rotor2 14 Schaufeln). Dieses Design wird als CRTF1 bezeichnet. Bild 5.4 zeigt, dass Memb 755 trotz reduzierter Schaufelzahl einen deutlich höheren Wirkungsgrad im ADP=OP0 aufweist. Obwohl die Kennlinie von Memb 755 von allen 3 Konfigurationen am flachsten verläuft und somit bei OP1 der kleinste Totaldruck aller Konfigurationen vorliegt, entspricht die erreichte Pumpgrenze den Anforderungen.

Zusätzlich ist durch die Forderung des hohen Wirkungsgrads im Approach Punkt OP2 ein verbessertes aeroakustisches Verhalten vom Memb755 zu erwarten.

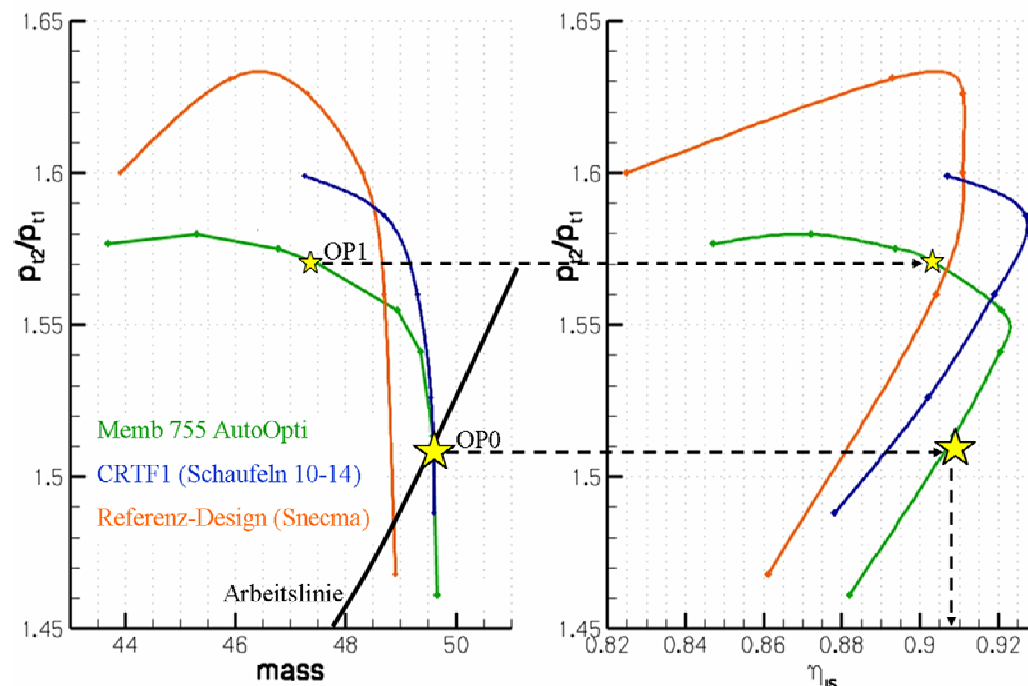


Bild 5.4: 100% Drehzahllinie der CRTF Konfigurationen orange: Referenz; blau: CRTF1; grün: AutoOpti Design:

Bild 5.5 und Bild 5.6 zeigen einen Vergleich der isentropen Machzahlverteilungen im ADP zwischen der initialen Auslegung (rechts) und Memb755 (links). Es zeigt sich, dass die Vorstoßmachzahlen in weiten Bereichen der Saugseite sowohl auf Rotor1 als auch auf Rotor2

deutlich reduziert wurden. Die damit verbundene Reduktion der Stoßverluste ist hauptverantwortlich für die Wirkungsgraderhöhung von Memb755 im ADP

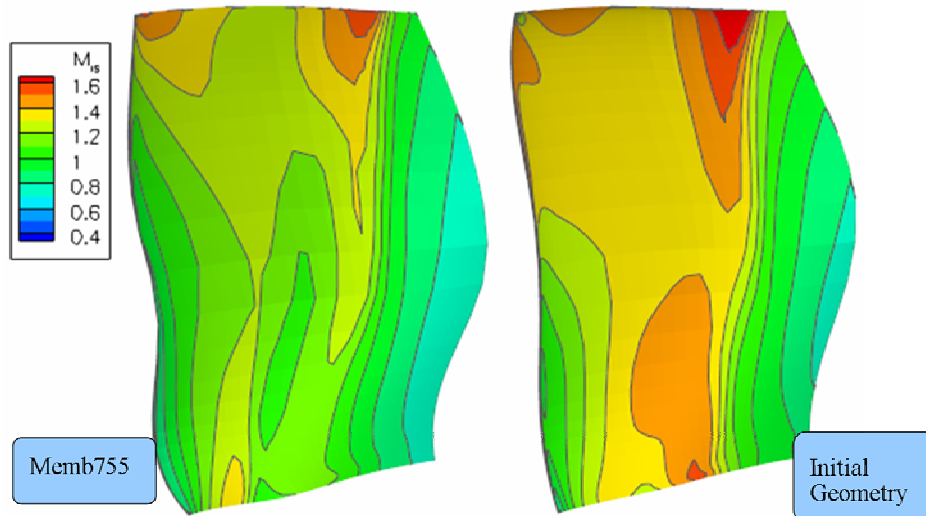


Bild 5.5: Vergleich der Machzahlverteilungen auf der Saugseite von Rotor1 im ADP.

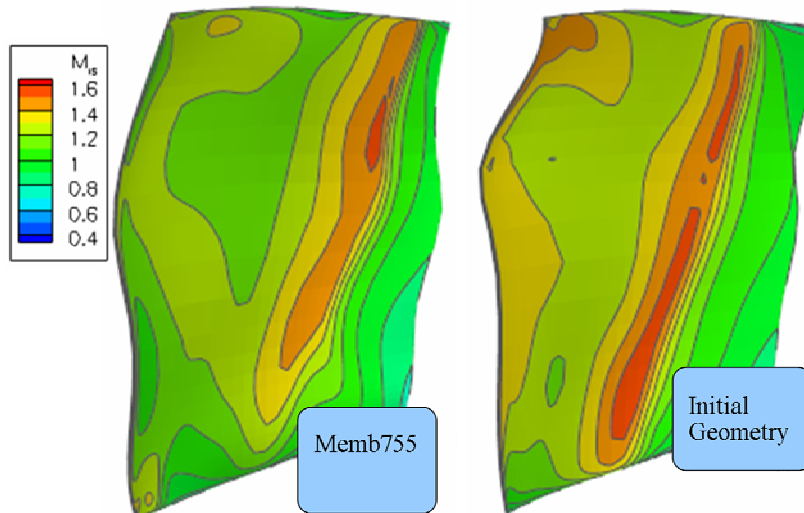


Bild 5.6: Vergleich der Machzahlverteilungen auf der Saugseite von Rotor2 im ADP.

Bild 5.7 zeigt die von Mises Spannungen vom Memb 755 auf Rotor1 und Rotor2. Die geforderte Restriktion, dass die maximale von Mises-Spannung sowohl für Rotor1 als auch für Rotor2 kleiner als 460 MPa bleibt ist durch Memb755 erfüllt. Diese maximalen Spannungen können auf beiden Schaufeln durch Umfangverschiebungen der Konstruktionsprofile noch deutlich reduziert werden. Eine Einbeziehung der Campbell-Diagramme kam in der vorliegenden Optimierung nicht zum Einsatz.

Zusammenfassung

Die hier vorgestellte Optimierung eines gegenläufigen Fans unter Einbeziehung mehrerer aerodynamisch relevanter Betriebspunkte und Berücksichtigung struktureller Nebenbedingungen verdeutlicht das Potential des entwickelten Programmpakets *AutoOpti*.

So ist es in diesem Beispiel trotz der relativ hohen Dimension des Suchraumes ($\text{dim} = 128$) gelungen, eine neue Fanstufe zu entwerfen die allen wesentlichen aerodynamischen Ansprüchen gerecht wird. Die hierfür benötigten 1950 TRACE-Rechnungen = 650 konvergente Member wurden auf 10 parallel laufenden Slave-Prozessen durchgeführt, denen jeweils 6 CPU's für die TRACE-Parallelisierung zur Verfügung standen (Insgesamt wurden also inklusive des Root-Prozesses 61 CPU's benötigt). Die Nutzung von Ersatzmodellen (hier: Kriging) zur Erzeugung viel versprechender Member konnte die Anzahl der benötigten Zielfunktionsaufrufe deutlich

reduzieren: Die Gesamtdauer der Optimierung betrug ca. 2,5 Wochen was auch für industrielle Maßstäbe akzeptabel erscheint.

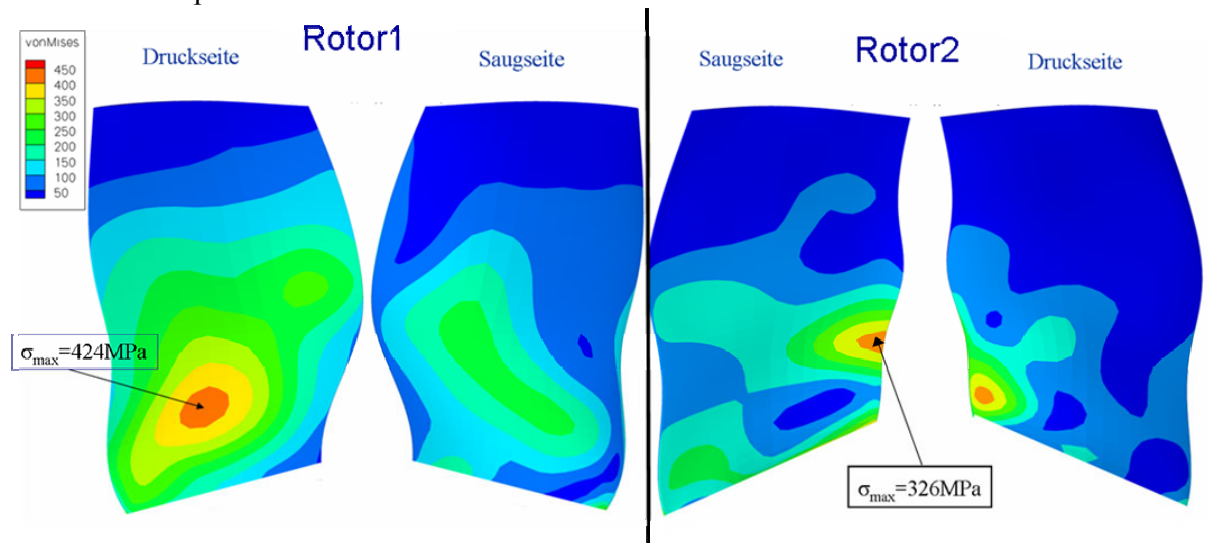


Bild 5.7 Van Mises-Spannungen von Memb 755: links Rotor1; rechts Rotor2

6. Optimierung mittels des gekoppelten Strömungslösers

Ein Optimierungsbeispiel welches den gekoppelten Löser TRACE-MAGELAN (siehe Kapitel 2.4.3) nutzt ist sowohl in der gemeinsamen Veröffentlichung DLR/Uni-Kassel [Becker et. al 2008] als auch im Abschlussbericht des COOREFF-Projektes 1.1.2 [Becker/Lawerenz 2008] detailliert beschrieben. Daher werden im Folgenden nur die wesentlichen Rahmenbedingungen und Ergebnisse vorgestellt.

Das Potential des Entwurfssystems, dessen aerodynamisches Modell die gekoppelte Strömungsrechnung darstellt, wird anhand des in [Becker/Lawerenz 2008] Kapitel 2.4 vorgestellten 3-stufigen Axialverdichters IDAC3 aufgezeigt. Für die Optimierung wurde hier der dritte Rotor, und als relevanter Betriebspunkt der ADP ausgewählt. Der gekoppelte Strömungslöser ist jedoch in der Lage mehrere Gitter dreidimensional zu berechnen, und es können in der Optimierung mehrere Betriebspunkte (siehe Kapitel 5) berücksichtigt werden. Es wurden in diesem Optimierungslauf 23 Designparameter im Bereich des dritten Rotors frei gegeben, die zum einen den Strakverlauf und zum anderen die Rotorgeometrie beeinflussen.

Der Zweck der Optimierung war in diesem Beispiel durch zwei Zielfunktionen definiert. Zum einen sollte der Wirkungsgrad maximiert werden. Daher wurde die erste Zielfunktion durch die Summe aus dem Wirkungsgrad des dritten Rotors und dem des Gesamtverdichters gebildet. Der Wirkungsgrad von Rotor 3 wurde dabei der 3D-Strömungssimulation von TRACE entnommen. Für den Gesamtwirkungsgrad wurden die Strömungsdaten von MAGELAN genutzt. Zum anderen sollte die aerodynamische Belastung des Verdichters reduziert werden. Die Analyse des Auslegungspunktes zeigte, dass der Stator 3 sehr hoch belastet ist. Die Diffusionszahlen betragen dort im Mittel $DF = 0,5061$. An der Nabe ergeben sich Maximalwerte von $DF_{\max} = 0,5524$. Aus diesem Grund wurden für den zweiten Zielfunktionswert die maximale und mittlere Diffusionszahl des letzten Stators summiert welche ebenfalls der Magelanrechnung entnommen wurden. Als zusätzliche Restriktion wurde verlangt, dass das Totaldruckverhältnis des Verdichters bei fester Drehzahl und vorgegebenem Massenstrom in engen Grenzen erhalten bleibt, so dass der Betriebspunkt unverändert bleibt.

Für die Verbesserung der Verdichterkonfiguration sind etwas mehr als 1000 Member ausgewertet worden. Die Optimierung wurde mit 30 Knoten ungefähr zwei Wochen durchgeführt. Zusätzlich wurden Antwortflächenmodelle auf zehn weiteren Rechnerknoten trainiert.

Nach Beendigung der Optimierung wurden 3 dominante Member Opti1, Opti2 und Opti3 der Paretofront ausgewählt. Die wesentlichen Maßzahlen sind in Tabelle 6.1 dargestellt. Opti1 zeichnet sich dabei durch einen hohen Wirkungsgrad aus, Opti3 zeigt eine deutliche Reduktion in der Belastung von Stator3 und Opti2 stellt einen Kompromiss dar.

	Opti1	Opti2	Opti3	Ref
Totaldruck π	2,0522	2,0139	2,0014	2,0174
Verdichter η	0,9038	0,9020	0,8978	0,8911
ROT3 η	0,9258	0,9237	0,9196	0,8987
DF _{max} Stat3	0,5773	0,5411	0,4859	0,5524
DF _{average} Stat3	0,5272	0,4923	0,4664	0,5061

Tabelle 6.1: Ergebnisse der TRACE-Magelan Optimierung des IDAC3

Bilder 6.1 zeigt einen Vergleich des initialen Verdichters (links) und der Wirkungsgrad-optimierten Konfiguration Opti1 (rechts) bzgl. der Entropieentwicklung im Gesamtverdichter (Magelan-Ergebnis, oben) und der Wirbelviskosität in der Rotor3 Passage (TRACE-Ergebnis, unten). Dabei zeigt sich, dass sowohl die Gesamtverdichterrechnung als auch die detaillierte 3D-CFD Einzelgitterrechnung eine deutliche Verbesserung vorhersagen

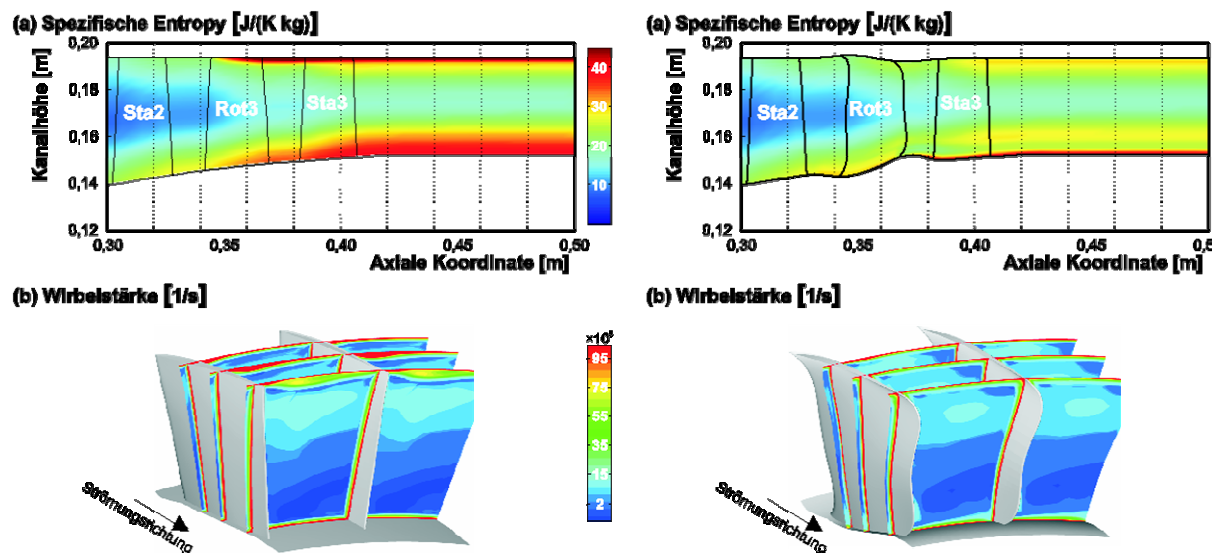


Bild 6.1: Entropie (oben) und Wirbelstärke (unten) der Referenz-Konfiguration (links) und Opti1-Konfiguration (rechts)

Dieses Optimierungsbeispiel zeigt, dass durch den in Kapitel 2.4.3 vorgestellten gekoppelten Strömungslöser die Anwendungsmöglichkeiten innerhalb einer Mehr-Ziel-Optimierung erweitert werden. Neben der Optimierung des dritten Rotors wird auch eine Verbesserung des Gesamtverdichterverhaltens erreicht. Denn die zugehörige Meridianströmungsrechnung erlaubt es, die Auswirkungen auf die benachbarten Schaufelreihen bzw. den Gesamtverdichter zu bewerten und in die Zielfunktionen einfließen zu lassen. Eine detaillierte Interpretation dieser positiven Ergebnisse findet sich ebenfalls in [Becker et. al 2008] und in [Becker/Lawerenz 2008]

7. Verwertung

Bereits während der Bearbeitungsphase dieses Vorhabens ist das Programmpaket *AutoOpti* in den aerodynamischen Auslegungszyklus von Axialverdichtern des Instituts für Antriebstechnik integriert worden. Es wird dort insbesondere mit den aerodynamischen Modellen Mises (zur Profilschnittoptimierung) und TRACE (3D Navier-Stokes Optimierung) eingesetzt (siehe Kapitel 2.4.1 und Kapitel 2.4.2). Durch den damit verbundenen regelmäßigen Gebrauch des hier entwickelten Programmpaketes in den unterschiedlichsten Forschungsvorhaben der Verdichterauslegung des Instituts, entsteht das notwendige Feedback für kontinuierliche Weiterentwicklungen und Verbesserungen. Des Weiteren wird dieses Programmpaket seit 2008 auch im DLR-Institut für Bauweisen- und Konstruktionsforschung in Stuttgart eingesetzt um strukturmechanische Optimierungen der Schwingungsmoden mittels des FEM-solvers Permas durchzuführen.

Der rasante Fortschritt im Bereich der Rechnerleistung und Simulationssoftware und die damit gewachsene wissenschaftliche und wirtschaftliche Akzeptanz automatischer Auslegungsmethoden, hat dazu geführt, dass *AutoOpti* mittlerweile in vielen nationalen und europäischen Vorhaben sowie in diversen Industrieaufträgen des DLR eingesetzt und weiter entwickelt wird. Nachfolgend sind einige Projekte genannt bei denen *AutoOpti* genutzt wird:

- **COORETEC-turbo; Vorhaben 1.2.1: Blattspitzenoptimierung (FKZ: 0327715 A)**

In diesem Vorhaben wird *AutoOpti* eingesetzt um gezielt den Blattspitzenbereich von subsonischen Rotoren stationärer Gasturbinenverdichter zu optimieren. Ziel ist es dabei den Spaltwirbel und die Randzonenverluste durch Geometrievariationen des Rotorblattes und des Gehäuses positiv zu beeinflussen. In Zusammenarbeit mit Siemens PG, dem Industriepartner in diesem Projekt, haben die erfolgreichen Optimierungsläufe hier schon zu einer gemeinsamen DLR-Siemens Patentanmeldung geführt.

- **COORETEC-turbo; Vorhaben 1.1.3: 3-D Rückführbeschaufelung (FKZ:0327715 O)**

Ziel dieses Vorhabens ist die Auslegung und Optimierung einer 3-D Radialverdichterschaufel mittels *AutoOpti*. Variiert werden sollen die Profilierung, die Meridionalkanalkontur und die Lage sowie der radiale Verlauf der Vorder- und Hinterkante. (Bisher 2-D Rückführbeschaufelung). Dieses Projekt wird zusammen mit dem Industriepartner MAN bearbeitet.

- **OPAL (Wirtschaftsförderung des Landes Brandenburg)**

Hier wird *AutoOpti* zur Mehrstufen Profilschnittoptimierung mittels TRACE (siehe Ende Kapitel 2.4.1) genutzt. Das Ziel dieses Vorhabens, welches in enger Zusammenarbeit mit Rolls-Royce Deutschland bearbeitet wird, ist es die Strutschaufeln eines Triebwerks so zu gestalten (umfangsabhängig), dass die Stomaufwirkung der Druckstörung, welche durch die nachfolgende bifurcation verursacht wird, minimiert wird. Hierdurch sollen möglichst rotationssymmetrische aerodynamische Randbedingungen für den Verdichter erreicht werden.

- **EU-Vorhaben Vital: Auslegung eines gegenläufigen Fans.**

Siehe hierzu Kapitel 5

- **EU-Vorhaben Dream: Auslegung eines gegenläufigen Propellers**

In diesem Vorhaben soll ein gegenläufiger Propeller ausgelegt werden. Die Ziele sind (ähnlich zu VITAL für ummantelte Rotoren), die Erhöhung des Wirkungsgrades bei hinreichendem Arbeitsbereich und geringem Lärm. Partner in diesem Vorhaben sind: Snecma, Onera, Cenaero, Ciam.

Neben dem Einsatz von *AutoOpti* in diesen Projekten und anderen Forschungsvorhaben mit DLR-Beteiligung, soll *AutoOpti* den an diesem Projekt beteiligten Industriepartner

MAN TURBO AG, Oberhausen
MTU Aero Engines GmbH, München
Siemens AG, Power Generation, Mülheim

zu marktüblichen Rahmenbedingungen angeboten werden. Daher haben bereits während der Bearbeitungsphase dieses Vorhabens zwei Übergabeseminare stattgefunden (in 2006 und 2007) in denen den Industriepartnern erste Testversionen von *AutoOpti* übergeben wurden.

Die Übergabe einer finalen Version von *AutoOpti* ist /Interesse der Industriepartner und Einigung über die Nutzungsmodalitäten vorausgesetzt) für Ende 2008 geplant.

Des Weiteren wurde mit dem COORETEC-turbo Tranche 3, Vorhaben 1.1.5; FKZ 0327717C eine Fortführung der erfolgreichen Entwicklungsarbeit an automatischen Optimierungsstrategien beantragt. Das Ziel dieses Vorhabens liegt in der Implementierung von instationären CFD-Simulationen und in der Implementierung von Gehäuseeinbauten (rotationssymmetrische und beliebige casing-treatements) in die automatische Prozesskette von *AutoOpti*. Mit dieser Erweiterung des Designraumes erhofft man sich eine deutliche Vergrößerung des Arbeitsbereiches zu erreichen ohne Einbußen im ADP-Wirkungsgrad hinnehmen zu müssen. Für dieses zukünftige Vorhaben konnte mit Rolls Royce Deutschland ein weiterer Industriepartner mit Interesse an einer Nutzung von *AutoOpti* gewonnen werden.

8. Zusammenfassung

Stationäre Gas- und Dampfturbinen sowie Flugtriebwerke haben eine zentrale Bedeutung für die heutige Welt. Kraftwerke mit Gas- und Dampfturbinen sind weltweit das Rückrad der elektrischen Energieversorgung. Der Luftverkehr nutzt nahezu ausschließlich Antriebe auf Turbomaschinenbasis. An der Dominanz der Turbomaschinen in der Energieversorgung und im Luftverkehr wird sich aller Voraussicht nach auch mittelfristig nichts ändern. Zentrale Ziele bei dem Entwurf moderner Triebwerke oder Gasturbinen sind die Absenkung des Brennstoffverbrauchs, des Eigengewichtes und der Baulänge sowie der Herstellungs-, Entwicklungs- und Wartungskosten.

Eine signifikante Leistungs- und Effizienzsteigerung von Turbomaschinenkomponenten erfordert die Ermittlung einer großen Zahl von geometrischen und physikalischen Parametern. Diese Parameter ergeben sich entweder direkt aus den spezifischen Anforderungen/Randbedingungen an die Komponente, oder müssen im Verlauf des Designprozesses der Komponente bzw. der gesamten Turbomaschine ermittelt werden.

Die enorme Komplexität der Verdichterauslegung hat, neben der bloßen Anzahl von zu ermittelnden Parametern, zwei wesentliche Ursachen:

- Interdisziplinarität (Aerodynamik, Strukturmechanik, Akustik, etc.)
- Mehrziel-Anforderungen pro Fachdisziplin

Diese große Anzahl von Zielen und Nebenbedingungen, welche sich aus den jeweiligen physikalisch-technischen Fachdisziplinen, und den Mehrziel-Anforderungen jeder einzelnen Fachdisziplin ergeben, sind im Allgemeinen konträr und müssen bewertet und gegeneinander abgewogen werden. Die Aufgabe besteht somit im Auffinden eines optimalen Kompromisses aus allen Anforderungen an das jeweilige Bauteil.

Aus diesem Grund ist in diesem Vorhaben das modular aufgebaute, parallelisierte Programmpaket *AutoOpti* entwickelt worden. Basierend auf der Evolutionsstrategie und beschleunigt durch die effiziente Nutzung von Ersatzmodellen (Kriging, neuronale Netze), erlaubt es prinzipiell Mehrzieloptimierungen beliebiger Optimierungsaufgaben, solange diese endlich dimensional parametrisierbar, und automatisch numerisch auszuwerten sind.

Die benötigte automatisierte Prozesskette für die Axialverdichterauslegung, von der Geometrieerzeugung über die Vernetzung bis hin zur überwachten CFD-Simulation ist ebenfalls innerhalb dieses Projektes aufgebaut worden. Somit ist durch die Arbeiten innerhalb des Forschungsvorhabens ein effizientes Werkzeug zur automatisierten aerodynamischen Auslegung vielstufiger Axialverdichter entstanden, dessen Einsatztauglichkeit in diversen Auslegungsbeispielen nachgewiesen werden konnte.

Die bisher durchgeführten Auslegungen decken den gesamten Designprozess von Axialverdichtern, wie er in Bild 2.11 dargestellt ist ab:

Die Throughflow-Optimierung ist wesentlicher Bestandteil des Partner-Vorhabens (COOREFF-T Vorhaben 1.1.2, Förderkennzeichen 0327720A) der Universität Kassel und im dazu gehörigen Abschlussbericht detailliert beschrieben ([Becker/Lawrenz 2008]). Ein ausführliches Beispiel für die Optimierung eines transsonischen Profilschnittes in mehreren Betriebspunkten mit Mises und *AutoOpti* findet sich in [Voß et al. 2006]. 3D-Navier-Stokes Optimierungen einer gegenläufigen Fanstufe unter Einhaltung strukturmechanischer Nebenbedingungen sind in Kapitel 5 beschrieben worden.

Durch die Kopplung von 2D (Throughflow) und 3D-Strömungsrechnungen eröffnen sich weitere Optionen zum Aufbau der Zielfunktionen. Neben den Strömungsgrößen der optimierten Gitter können nun auch die das Gesamtverhalten beschreibenden Kenngrößen wie z.B. der Gesamtwirkungsgrad verwendet werden, um die einzelnen Kandidaten der Optimierung zu bewerten (siehe Kapitel 6 und [Becker et. al 2008]).

In zukünftigen Arbeiten (geplantes Vorhaben: COORETEC-turbo Tranche 3, Vorhaben 1.1.5; FKZ 0327717C) soll diese erfolgreichen Arbeiten weiter geführt werden und *AutoOpti* auf andere Turbomaschinen Komponenten (z.B. Radialverdichter, Gehäuseeinbauten) und physikalische Disziplinen (instationäre CFD) erweitert werden.

Literaturverzeichnis

[Abbas et al. 2001] Abbas, H. A., R. Sarker und C. Newton (2001). PDE: A Pareto-Frontier Differential Evolution Approach For Multi-Objective Optimization Problems. Proceedings Of The Congress On Evolutionary Computation, 2:971–978.

[Adamczyk et al. 1986] Adamczyk, J.J., R. Mulac und M. Celestina (1986). A Model for Closing the Inviscid Form of the Average-Passage Equation System. ASME Paper 86-GT-227, ASME Gas Turbine Conference, Düsseldorf, Germany, June 9-12, 1986.

[Ahmed 2001] Ahmed, R. (2001). Parallele Optimierungsalgorithmen zur Auslegung vielstufiger Axialverdichter . GuD-Kraftwerk 500 MW auf einer Welle, Abschlussbericht, Vorhaben 1.2.7, AG Turbo.

[Ahmed 2005] Ahmed, Raffael (2005). Auslegung vielstufiger Axialverdichter mit parallelen Evolutionsstrategien und neuronalen Netzen, Bd. Reihe 7, Nr. 469. VDI Verlag.

[Ahmed und Lawerenz 2003] Ahmed, Raffael und M. Lawerenz (2003). On The Aero-Mechanical Design Of Multistage Axial Compressors Using Parallel Optimization Algorithms. In: International Symposium On Air Breathing Engines 2003, Nr. ISABE 2003-1017.

[Aurenhammer 1991] Aurenhammer, Franz (1991). Voronoi Diagrams – A Survey Of A Fundamental Geometric Data Structure. ACM Computing Surveys, 23(3):345–405.

[Backhaus 2007] Automatische Erzeugung von Rechnernetzen für Finite-Elemente Rechnungen Institutsbericht des Instituts für Antriebstechnik, DLR Köln, IB-325-07-07

[Backhaus 2007 b] Integration einer Finite-Elemente Strukturanalyse in die Prozesskette der automatischen Verdichteroptimierung; Institutsbericht des Instituts für Antriebstechnik, DLR Köln, IB-325-16-07

[Baier 1998] Baier, Ralf-Dietmar (1998). EVOTECH - Einsatz der Evolutionsstrategie in Wissenschaft und Technik. FKZ 01 IB 404 D0, Daimler-Benz Aerospace, MTU.

[Baralon 2000] Baralon, S. (2000). On Multistage Analysis Of Transonic Compressors: From Axisymmetric Throughflow Time-Marching To Unsteady Three-Dimensional Methods. Doktorarbeit, Chalmers University of Technology, Göteborg.

[Baralon et al. 1997] Baralon, S., L.-E. Eriksson und U. Hall (1997). Viscous Throughflow Modelling Of Transonic Compressors Using A Time-Marching Finite Volume Solver . In: 13th International Symposium On Air Breathing Engines (ISABE), Chattanooga, USA.

[Bauermeister 1963] Bauermeister, K. J. (1963). Probleme der Sekundärströmungen in axialen Turbomaschinen, Bd. VDI-Forschungsheft 496, Kap. Teil II: über den Einfluß der Schaufelbelastung auf die Sekundärströmung in geraden Turbinen- und Verdichtergittern. VDI-Verlag.

[Bäck und Schwefel 1996] Bäck, Thomas und H.-P. Schwefel (1996). Evolutionary Computation: An Overview. In: Proceedings Of The Third IEEE Conference On Evolutionary Computation 1996, S. 20–29. IEEE-Press.

[Becker et. al 2008] K. Becker, M. Lawerenz, C. Voß, R. Mönig: MULTI-OBJECTIVE OPTIMIZATION IN AXIAL COMPRESSOR DESIGN USING A LINKED CFD-SOLVER, ASME International Gas Turbine & Aeroengine Technical Congress & Exposition, 2008

[Becker/Lawerenz 2008] K. Becker, M. Lawerenz: Abschlussbericht zum COOREFF-T Vorhaben 1.1.2; Automatische Optimierung von Verdichterschaufeln, Förderkennzeichen 0327720A

[Benini und Toffolo 2001] Benini, Ernesto und A. Toffolo (2001). Systematic Two-Dimensional Cascade Optimization For Axial Flow Pumps Using Genetic Algorithms. In: 4th European Conference On Turbomachinery, Fluid Dynamics And Thermodynamics, Nr. ATI-CST-037/01.

[Bohne 2002] Bohne, A. (2002). Einfluss von instationärer Strömung und Turbulenz auf die Profilumströmung und das Grenzschichtverhalten moderner Beschaufelungen mehrstufiger Axialverdichter im off-design-Bereich. Abschlussbericht Vorhaben 694, Forschungsvereinigung Verbrennungskraftmaschinen e. V.

[Bohne und Niehuis 2004] Bohne, A. und R. Niehuis (2004). Experimental Off-Design Investigation of Unsteady Secondary Flow Phenomena in a Three-Stage Axial Compressor at 68% Nominal Speed. ASME Paper GT2004-5310.

[Box und Wilson 1951] Box, George E. P. und K. B. Wilson (1951). On The Experimental Attainment Of Optimum Conditions. Journal of the Royal Statistical Society, 13(1):1–45.

[C, Cetin et al. 1987] C, Cetin, M., A. Ücer, C. Hirsch und G. Serovy (1987). Application of Modified Loss and Deviation Correlations to Transonic Axial Compressors. AGARD Report 745, Advisor Group For Aerospace Research And Development, Neuilly-Sur-Seine, France.

[Chung und Alonso 2004] Chung, Hyoung-Seog und J. J. Alonso (2004). Multiobjective Optimization Using Approximation Model-Based Genetic Algorithms. In: 10th AIAA/ISSMO Symposium on Multidisciplinary Analysis and Optimization, Nr. AIAA 2004-4325.

[Cumpsty 2004] Cumpsty, N. A. (2004). Compressor Aerodynamics. Krieger Publishing Company; 2nd Revised Edition.

[Damle et al. 1995] Damle, S.V., T. Dang und D. Reddy (1995). Throughflow Method For Turbomachines Applicable For All Flow Regimes. ASME Paper 95-GT-395, International Gas Turbine and Aeroengine Congress and Exposition Houston, Texas - June 5-8, 1995.

[Dennis 1999] Dennis, B. H.; Dulikravich, G. S.; Han, Z.-X.: Constrained Shape of Optimization of Airfoil Cascades using a Navier-Stokes Solver and a Genetic/SQP Algorithm. ASME 99-GT-441, 1999.

[Dorfner et al. 2007] Dorfner C., Voß C. Nicke E.: AXIS-ASYMMETRIC PROFILED ENDWALL DESIGN USING MULTIOBJECTIVE OPTIMIZATION LINKED WITH 3D RANS-FLOW-SIMULATIONS, ASME International Gas Turbine & Aeroengine Technical Congress & Exposition, 2007

[Drela 1998] Mark Drela, Harold Youngren: A User's guide to Mises MIT Computational Aerospace Sciences Laboratory, 1998

[Emmerich et al. 2003] Emmerich, Michael, A. Giotis, M. Özdemir, T. Bäck und K. Giannakoglou (2003). Lecture Notes In Computer Science, Kap. Metamodel-Assisted Evolution Strategies, S. 361–370.

- [Fay 2002] Fay, Gudrun (2002). Zur Berechnung der Meridianströmung subsonischer Axialverdichter auf der Basis der umfangsgemittelten Navier-Stokes-Gleichungen. Dissertation, Universität Kassel, Fachgebiet Strömungsmaschinen. Shaker-Verlag.
- [Fogel 1994] Fogel, D. B., "An Introduction to Simulated Evolutionary Optimization", IEEE Trans. on Neural Networks: Special Issue on Evolutionary Computation, Vol. 5, No. 1, pp. 3-14, 1994.
- [Gauger 2006] Gauger, Nicolas R. (2006). Adjoint Approaches In Aerodynamic Shape Optimization And MDO Context. In: Lecture Series 2006-03. von Karman Institute for Fluid Dynamics.
- [Giannakoglou und Karakasis 2006] Giannakoglou, Kyriakos C. und M. K. Karakasis (2006). Hierarchical And Distributed Metamodel-Assisted Evolutionary Algorithms. In: Lecture Series 2006-03. von Karman Institute for Fluid Dynamics.
- [Giannakoglou und Papadimitriou 2006] Giannakoglou, Kyriakos C. und D. I. Papadimitriou (2006). Formulation And Application Of The Continuous Adjoint Method In Aerodynamics And Turbomachinery. In: Lecture Series 2006-03. von Karman Institute for Fluid Dynamics.
- [Giotis und Giannakoglou 1999] Giotis, A.P. und K. Giannakoglou (1999). Single- And Multi-Objective Airfoil Design Using Genetic Algorithms And Artificial Intelligence. In: EUROGEN99 - Evolutionary Algorithms In Engineering And Computer Science.
- [Glas und Jaberg 2001] Glas, Wolfgang und H. Jaberg (2001). Multi-Objective Evolutionary Algorithm For The Optimization Of Swept Pump Impellers. In: 4th European Conference On Turbomachinery, Fluid Dynamics and Thermodynamics, Nr. ATI-CST-038/01, S. 469–479.
- [Grieb et al. 1975] Grieb, H., G. Schill und R. Gumucio (1975). A Semi-Empirical Method For The Determination Of Multistage Axial Compressor Stage Efficiency.
- [Hammel und Bäck 1994] Hammel, Ulrich und T. Bäck (1994). Evolution Strategies On Noisy Functions: How To Improve Convergence Properties. In: Parallel Problems Solving From Nature - PPSN III , S. 418–427. International Conference on Evolutionary Computation. Springer Verlag.
- [Holland 1975] Holland, J. H., "Adaptation in natural and artificial systems", Ann Arbor: The University of Michigan Press, 1975.
- [Hübner und Fottner 1996] Hübner, J. und L. Fottner (1996). Influence of Tip-Clearance, Aspect Ratio, Blade Loading, and Inlet Boundary Layer on Secondary Losses in Compressor Cascades. ASME paper 96-GT-505, ASME. International Gas Turbine and Aeroengine Congress and Exhibition Birmingham, UK - June 10-13, 1996.
- [Hirsch und Denton 1981] Hirsch, Ch. und J. Denton (1981). Throughflow Calculations In Axial Turbomachines. AGARD Advisory Report, (175).
- [Hoynacki 1999] Hoynacki, A. (1999). Einfluss von instationärer Strömung und Turbulenz auf die Grenzschicht und auf die Druckverteilungen von Beschaufelungen moderner mehrstufiger Verdichter . Abschlussbericht Vorhaben Nr. 601, Forschungsvereinigung Verbrennungskraftmaschinen e.V.
- [Jameson et al. 1981] Jameson, A., W. Schmidt und E. Turkel (1981). Numerical Solution Of The Euler Equations By Finite Volume Methods Using Runge Kutta Time Stepping Schemes. AIAA Paper 81-1259, S. 1–13.

[Keane 2006] A.J. Keane: Statistical improvement criteria for use in multi-objective design optimization. IMechE Seminar Proceedings, June 2006

[Koza 1992] John R. Koza: Hierarchical Automatic Function Definition in Genetic Programming. FOGA 1992: 297-318

[Kügeler et al. 2001] Kügeler, E., A. Weber und S. Lisiewicz (2001). Combination Of A Transition Model With A Two-Equation Turbulence Model and Comparison With Experimental Results. Proceedings Of The 4th European Turbomachinery Conference, Florence, Italy, ATI-CST- 076/01.

[Lawerenz et al. 2002] Lawerenz, M., L. Förster und G. Fay (2002). Entwicklung eines Verfahrens zur Berechnung der reibungsbehafteten Meridianströmung auf der Basis umfangsgemittelter Navier-Stokes-Gleichungen. Abschlussbericht Vorhaben Nr. 689, Heft 737, Forschungsvereinigung Verbrennungskraftmaschine e. V., Frankfurt.

[Lawerenz 1993] Lawerenz, Martin (1993). Randströmung in Axialverdichtern. In: Tagungsband der DGLR-Fachausschußsitzung "Dreidimensionale, reibungsbehaftete Strömung in Turbomaschinen" DGLR, Köln-Porz.

[Lawerenz 1995] Lawerenz, Martin (1995). Aerodynamische Optimierung von Strömungsmaschinen unter Verwendung direkter numerischer Optimierungsverfahren. Abschlußbericht, TURBOTECH-Verbundvorhaben 1.1.2.13 MTUM - B95EW-0014, MTU München GmbH.

[Lieblein 1965] Lieblein, Seymour (1965). Aerodynamic Design Of Axial Flow Compressors Kap. VI. Experimental Flow in Two-Dimensional Cascades, S. 183–226. National Advisory Committee For Aeronautics, Washington. [Madavan 2002] Madavan, N. K. (2002). Multi-Objective Optimization Using A Pareto Differential Evolution Approach. Proceedings Of the Congress on Evolutionary Computation, 2:1145–1150.

[Müller-Töws 2000] Müller-Töws, Jörg (2000). Aerothermodynamische Auslegung der Meridianströmung mehrstufiger Axialverdichter mit Hilfe von Optimierungsstrategien. Dissertation, Fachbereich Maschinenbau der Universität Kassel.

[Moore 1985] Moore, J. G. (1985). An Elliptic Procedure For 3D Viscous Flow. In: 3D Computation Techniques Applied To Internal Flows In Propulsion Systems. AGARD LS-140.

[Myers und Montgomery 2002] Myers, Raymond H. und D. C. Montgomery (2002). Response Surface Methodology. Wiley Interscience.

[Naujoks et al. 2000] Naujoks, Boris, L. Willmes, W. Haase, T. Bäck und M. Schütz (2000). Multi-Point Airfoil Optimization Using Evolution Strategies. In: European Congress on Computational Methods in Applied Sciences and Engineering.

[Niehuis et al. 2003] Niehuis, R., A. Bohne und A. Hoynacki (2003). Experimental Investigation of Unsteady Flow Phenomena in a Three Stage Axial Compressor . In: 5th European Conference on Turbomachinery, Prague.

[Nissen 1997] V. Nissen: Einführung in Evolutionäre Algorithmen. Optimierung nach dem Vorbild der Evolution, Computational Intelligence, Vieweg Verlag, 1997.

- [Nürnberg 2004] Nürnberg, D. (2004). Implizite Zeitintegration für die Simulation von Turbomaschinenströmungen. Dissertation, Fakultät für Maschinenbau der Ruhr-Universität Bochum.
- [Numeca 2006] Numeca (2006). FINE/Turbo v7 (Including Euranus) - Flow Integrated Environment. User Manual, Numeca International.
- [Okabe et al. 2000] Okabe, Atsuyuki, B. Boots, K. Sugihara und S. N. Chiu (2000). Spatial Tesselations: Concepts And Applications Of Voronoi Diagrams. John Wiley & Sons.
- [Oyama 2000] Oyama, Akira (2000). Multidisciplinary Optimization Of Transonic Wing Design Based On Evolutionary Algorithms Coupled With CFD Solver . In: European Congress On Computational Methods In Applied Sciences And Engineering.
- [Papila et al. 2002] Papila, Nilay, W. Shyy, L. Griffin und D. Dorney (2002). Shape Optimization of Supersonic Turbines Using Global Approximation Methods. Journal of Propulsion and Power, 18(3):509–518.
- [Petrovic et al. 2000] Petrovic, Milan V., G. S. Dulikravich und T. J. Martin (2000). Maximizing Multistage Turbine Efficiency by Optimizing Hub And Shroud Shapes And Inlet And Exit Conditions Of Each Blade Row. International Journal of Turbo Jet Engines, 17(4):267–278.
- [Piegl und Tiller 1997] Piegl, L. und W. Tiller (1997). The NURBS Book, Monographs in Visual Communication. 2nd ed., ISBN 3-540-61545-8, Springer Verlag.
- [Pierret und van den Braembussche 1999] Pierret, S. und R. van den Braembussche (1999). Turbomachinery Blade Design Using A Navier-Stokes Solver And Artificial Neural Network. Journal of Turbomachinery, 121:326–332.
- [Press et al. 2002] Press, William H., S. A. Teukolsky, W. T. Vetterling und B. P. Flannery (2002). Numerical Recipes In C - The Art Of Scientific Computing, 2nd Edition. Cambridge University Press.
- [Periaux et al. 2000] P'eriaux, Jacques, G. Degrez und H. Deconinck (2000). Genetic Algorithms For Optimisation In Aeronautics And Turbomachinery. VKI Lecture Series 2000-07, Karman Institute for Fluid Dynamics.
- [Price et al. 2005] Price, Kenneth V., R. M. Storn und J. A. Lampinen (2005). Differential Evolution. A Practical Approach to Global Optimization. Springer-Verlag Berlin Heidelberg
- [Rai 2006] Rai, M. M. (2006). Single- and Multiple-Objective Optimization with Differential Evolution and Neural Networks. In: Lecture Series 2006-03. von Karman Institute for Fluid Dynamics.
- [Rechenberg 1973] Rechenberg, I., „Evolutionsstrategie - Optimierung technischer Systeme nach Prinzipien der biologischen Evolution“, Stuttgart: Frommann-Holzboog, 1973.
- [Rechenberg 1994] Rechenberg, Ingo (1994). Evolutionsstrategie '94. Frommann-Holzboog.
- [Rhie 1986] Rhie, C. M. (1986). A Pressure Based Navier-Stokes Solver Using the Multigrid Method.
- [Saka et al. 2007] Saka, Yuki, M. Gunzburger und J. Burkardt (2007). Latinized, Improved LHS, And CVT Point Sets In Hypercubes. International Journal of Numerical Analysis and Modeling, 4(3-4):729–743.

[Sasaki et al. 2001] Sasaki, D., S. Obayashi und H.-J. Kim (2001). Evolutionary Algorithm vs. Adjoint Method Applied to SST Shape Optimization. In: The Annual Conference of CFD Society of Canada, Waterloo, Canada, May 27 - 29.

[Schittkowski 1985] Schittkowski, K. (1985). NLPQL: A Fortran subroutine for solving constrained nonlinear programming problems. *Annals of Operations Research*, 5:485–500.

[Schmitz 2008] Entwicklung eines Softwaretools zur Analyse des dynamischen Verhaltens von Triebwerksschaufeln, Praxissemesterbericht der Fachhochschule Köln, Fakultät für Informatik und Ingenieurwissenschaften absolviert beim DLR e.V. Kön-Porz: Institut für Antriebstechnik

[Schwefel 1995] Schwefel, Hans-Paul (1995). *Evolution and Optimum Seeking*. John Wiley & Sons.

[Shahpar 2000] Shahpar, Shahrokh (2000). A Comparative Study of Optimisation Methods for Aerodynamic Design of Turbomachinery Blades. In: *Proceedings of ASME TURBOEXPO*, Nr. 2000-GT-523.

[Shyy et al. 2001] Shyy, Wei, N. Papila, R. Vaidyanathan und K. Tucker (2001). Global Design Optimization For Aerodynamics And Rocket Propulsion Components. In: *Progress in Aerospace Sciences*, Bd. 71, S. 59–118.

[Simon 2007] Simon, Jean-François (2007). *Contribution To Throughflow Modelling For Axial Flow Turbomachines*. Dissertation, Université de Liège.

[Spiegel und Rist 1995] Spiegel, Michael-Stefan und D. Rist (1995). Direkte numerische Optimierungsverfahren zur Lösung aerodynamischer Problemstellungen bei Turbomaschinen. Abschlussbericht, TURBOTeCH-Verbundvorhaben 1.1.2.14, TU München, Institut für Luft- und Raumfahrt.

[Spurr 1980] Spurr, A. (1980). The Prediction Of 3D Transonic Flow In Turbomachinery Using A Combined Throughflow And Blade-To-Blade Time Marching Method. *The International Journal of Heat and Fluid Flow*, 2(4):189–199.

[Storn und Price 1996] Storn, Rainer und K. Price (1996). *IEEE International Conference On Evolutionary Computation*, Nagoya, May 1996, Kap. Minimizing The Real Functions Of The IEC'96 Contest By Differential Evolution, S. 842 – 844. IEEE, New York.

[Sturmayr 2004] Sturmayr, Andreas (2004). *Evolution of a 3D Structured Navier-Stokes Solver Towards Advanced Turbomachinery Applications*. Dissertation, Vrije Universiteit Brussel.

[Thompson et al. 1985] Thompson, J.F., Z. Warsi und C. Mastin (1985). *Numerical Grid Generation*. North Holland.

[Trigg et al. 1997] Trigg, M. A.; Tubby, G. R.; Sheard, A. G.: Automatic Genetic Optimization Approach to 2D Blade Profile Design for Steam Turbines. *ASME 97-GT-392*, 1997.

[Trigg et al. 1999] Trigg, M. A., G. R. Tubby und A. G. Sheard (1999). Automatic Genetic Optimization Approach To Two-Dimensional Blade Profile Design For Steam Turbines. *Journal Of Turbomachinery*, 121:11–17.

[Uelschen 2000] Uelschen, Michael (2000). Entwurf und Optimierung der zweidimensionalen Gitterströmung axialer Turbomaschinenbeschaufelungen mit Neuronalen Netzen und Genetischen Algorithmen. Dissertation, Universität Kassel, Fachgebiet Strömungsmaschinen.

[Vaidyanathan et al. 2000] Vaidyanathan, Rajkumar, N. Papila, W. Shyy, K. Tucker, L. Griffin, R. Haftka und N. Fitz-Coy (2000). Neural Network and Response Surface Methodology for Rocket Engine Component Optimization. AIAA 2000-4880.

[Voß et al. 2006] Voß, Christian, M. Aulich, B. Kaplan und E. Nicke (2006). Automated Multiobjective Optimisation In Axial Compressor Blade Design. ASME Paper GT2006-90420, 51th ASME International Gas Turbine & Aeroengine Technical Congress & Exposition, Barcelona, Spain, May 8-11, 2006.

[Waldtke 1995] Waldtke, U. (1995). Berechnung des Meridianströmungsfeldes in Axialverdichtern mit der Methode der Finiten Elemente. Technischer Bericht 261, VDI-Fortschrittsberichte.

[Watzlawick und Fottner 1992] Watzlawick, R. und L. Fottner (1992). Bestimmung der wesentlichen Einflußgrößen in der Korrelation der Sekundärströmungsverluste bei Veränderung des Schaufelseitenverhältnisses. Motortechnische Zeitschrift, 53:276–284.

[Weber 2004] Weber, A. (2004). 3D Structured Grids for Multistage Turbomachinery Applications based on G3DMESH. 1st revision ed. Institute of Propulsion Technology, German Aerospace Centre, Cologne, February, DLR IB-325-05-04.

[Wu 1952] Wu, C.-H. (1952). A General Theory Of Three-Dimensional Flow In Subsonic And Supersonic Turbomachines Of Axial- Radial- And Mixed-Flow Types. Technischer Bericht Technical Note 2604, National Advisory Committee for Aeronautics (NACA).

[Zell 1997] Zell, Andreas (1997). Simulation neuronaler Netze. Oldenbourg Verlag.